Università di Pisa
Dipartimento di Matematica
Corso di Dottorato in Matematica

Computational Aspects of Line and Toric Arrangements

Tesi di Dottorato

Candidato
Oscar Papini

Relatori
Prof. **Mario Salvetti**
Prof. **Giovanni Gaiffi**

XXX Ciclo (2014–2017)

Abstract

In the first part of this thesis we deal with the theory of hyperplane arrangements, that are (finite) collections of hyperplanes in a (finite-dimensional) vector space. If $\mathcal{A}$ is an arrangement, the main topological object associated with it is its *complement* $\mathcal{M}(\mathcal{A})$ (Definition 1.6), and the main combinatorial object is its *intersection poset* $\mathrm{L}(\mathcal{A})$ (Definition 1.10). Many studies have been done in order to understand what topological properties of $\mathcal{M}(\mathcal{A})$ can be inferred from the knowledge of the combinatorial data of $\mathrm{L}(\mathcal{A})$.

In the thesis we focus on the *characteristic variety* $\mathcal{V}(\mathcal{A})$ associated with an arrangement $\mathcal{A}$ (Definition 3.1), in an effort to uncover some possible combinatorial description of it. Characteristic varieties have been studied for some years and information about them could shed more light on other topological objects, such as the so-called Milnor fibre of an arrangement. Unfortunately there are not many examples of computed characteristic varieties in the literature, because the algorithms involved require many computational resources and are both time- and memory-consuming. To overcome this problem, we developed some new algorithms that are able to compute characteristic varieties for general arrangements. In Chapter 4 we describe them in full details, together with actual `SageMath` code, so that other researchers could follow our path; in Chapter 5 we collect the results in a little catalogue. We tried to evince some general combinatorial pattern from them (see Section 3.7), but we leave our considerations in conjecture form.

The second part of the thesis is focused on toric arrangements, which are finite collections of subtori (called *layers* in this context) in the complex algebraic torus. In particular, we follow two articles by De Concini and Gaiffi [11, 12] in which they compute *projective wonderful models* for the complement of a toric arrangement (Definition 6.13) and a presentation of their integer cohomology rings. Also in this case we develop an algorithm that is able to produce examples of such rings (Chapter 7). This is a little step, but we hope that the possibility to compute more examples, together with better and more efficient algorithms, can greatly improve the understanding of these topological objects.

Contents

List of Symbols

$\#(A)$	cardinality of the set A
A^c	complement of the set A
$\wp(A)$	power set of the set A
$\simeq$	is isomorphic to
$[n]$	$=\{1,\dots,n\}$, set of integers between 1 and n

Spaces and Algebraic Structures

$V^\vee$	dual space of the vector space V
$\langle W\rangle$	span of the set W; substructure generated by the set W
$\mathcal{M}_{m\times n}(R)$	set of $m\times n$ matrices with coefficients in R
A^T	transposed of the matrix A
$\mathcal{H}om(M,N)$	homomorphisms between two algebraic structures M and N
$\mathcal{H}om_R(M,N)$	R-linear homomorphisms between two R-modules M and N
$[G,G]$	commutator subgroup of the group G
RG	group ring of the group G with coefficients in the ring R
R^*	set of invertible elements of the ring R
(S)	ideal generated by the subset S
$\sqrt{I}$	radical of the ideal I
$R[T^{\pm 1}]$	ring of univariate Laurent polynomials in T with coefficients in R
$\mathcal{Z}(I)$	zero locus of the ideal I
D^n	unit disk of dimension n
S^n	unit sphere of dimension n
$T_p(X)$	tangent space of the variety X at a non-singular point p

Hyperplane Arrangements

$Q_{\mathcal{A}}(\mathbf{X})$	defining polynomial of $\mathcal{A}$
$\mathcal{M}(\mathcal{A})$	complement of $\mathcal{A}$

(continued on next page)

$\cup\mathcal{B}$	union of the hyperplanes of $\mathcal{B}$
$\cap\mathcal{B}$	intersection of the hyperplanes of $\mathcal{B}$
$L(\mathcal{A})$	intersection poset of $\mathcal{A}$
$A(\mathcal{A})$	Orlik-Solomon algebra of $\mathcal{A}$
$c\mathcal{A},\ d\mathcal{A}$	coning, deconing of $\mathcal{A}$
$\overline{\mathcal{A}},\ \widehat{\mathcal{A}}$	projectivization, deprojectivization of $\mathcal{A}$
$\mathcal{S}ing(\mathcal{A})$	set of singular points of the line arrangement $\mathcal{A}$
$\mathcal{A}_{\mathbb{C}}$	complexification of $\mathcal{A}$
$\mathcal{V}^i_k(\mathcal{A})$	characteristic variety of $\mathcal{A}$ of degree i and depth k
$\check{\mathcal{V}}_k(\mathcal{A})$	"homogeneous part" of the characteristic variety
$\mathcal{R}^i_k(\mathcal{A})$	resonance variety of $\mathcal{A}$ of degree i and depth k
$\mathcal{I}(P), \mathcal{I}(\pi)$	ideal associated with the singular point P, the partition π
$\mathcal{B}r_n$	braid arrangement in $\mathbb{K}^n$

Toric Arrangements

$X^*(T)$	group of characters of the torus T
$X_*(T)$	group of one-parameter subgroups of the torus T
$C(\mathbf{r}_1, \ldots, \mathbf{r}_h)$	cone generated by $r_1, \ldots, r_h$
X_Δ	toric variety associated with the fan Δ
$\mathcal{C}(\mathcal{A})$	poset of layers of a toric arrangement $\mathcal{A}$

Abbreviations

$p\mathcal{A}$, $a\mathcal{A}$	$= \overline{c\mathcal{A}},\ d\widehat{\mathcal{A}}$
$\mathcal{V}_k(\mathcal{A})$	$= \mathcal{V}^1_k(\mathcal{A})$
$\mathcal{V}(\mathcal{A})$	$= \mathcal{V}_1(\mathcal{A})$
$\mathcal{R}_k(\mathcal{A})$	$= \mathcal{R}^1_k(\mathcal{A})$
$\mathcal{R}(\mathcal{A})$	$= \mathcal{R}_1(\mathcal{A})$

Notational Remark. In this work, **boldface** symbols (such as $\mathbf{a}$) represent multi-dimensional objects, like vectors or n-tuples, whereas roman symbols (such as a) usually denote one-dimensional values. Variables or unknown quantities are written with lower case letters, while upper case is used for indeterminates in polynomial rings.

Introduction

It is easy to explain what a hyperplane arrangement is, even to a child, at least in low dimension: just draw a bunch of lines on a sheet of paper. Despite their apparent innocence, though, hyperplane arrangements have been keeping busy mathematicians for some 60 years, and not everything about them is well-understood.

The theory of hyperplane arrangements lies in the intersection of several branches of mathematics, mainly combinatorics, algebra and topology. A major question is how the different aspects of an arrangement interact with each other. In particular, we would like to know what algebraic and topological properties of an arrangement depend only on the combinatorial data associated with it. These data are encoded in the *intersection poset* $L(\mathcal{A})$ of an arrangement $\mathcal{A}$: it is the poset whose elements are all the non-empty intersections of some hyperplanes of the arrangement, partially ordered by reverse inclusion (Definition 1.10).

This question does actually make sense, because there are examples of important algebraic and topological objects that fall in both categories. The cohomology ring of the complement of an arrangement *does* depend only on the intersection poset: it is isomorphic to the Orlik-Solomon algebra, which is defined in terms of $L(\mathcal{A})$. On the other hand, the fundamental group of the complement *does not* depend only on $L(\mathcal{A})$: Rybnikov [39] successfully built a pair of arrangements which have isomorphic intersection posets, but non-isomorphic fundamental groups of the complements.

Among the topological objects for which it is still unknown whether they are combinatorially determined, we chose to study the *characteristic variety* $\mathcal{V}(\mathcal{A})$ of an arrangement. It is defined as the jumping locus of the cohomology with local coefficients of the complement of the arrangement (Definition 3.1), and can be viewed as a subset of the complex torus $(\mathbb{C}^*)^n$, where n is the number of hyperplanes of the arrangement. It is known that it is an algebraic variety: to be more precise, Arapura's Theorem [1] states that it is a finite union of possibly translated subtori of $(\mathbb{C}^*)^n$.

The study of the characteristic varieties is significant *per se*, but a better understanding of their properties could shed light also on open problems related to other topological objects. For example, the homology of the Milnor fibre can be retrieved from the characteristic variety, together with its monodromy action. Several authors are actively searching for how much information about the Milnor fibre is possible to induce from the combinatorics, see for example [3, 42, 47, 48].

We know that the irreducible components of the characteristic variety are algebraic tori, thanks to Arapura's Theorem; some of them pass through the point $(1, \dots, 1)$ and some don't: the former are "homogeneous" components, the latter are "translated" ones. Translated components do exist indeed, and the first 1-dimensional one was found in [46]. It is known that the homogeneous part of the characteristic variety *is* combinatorially determined, because the homogeneous components are in 1-1 correspondence with the components of the *resonance variety* $\mathcal{R}(\mathcal{A})$, which is defined in terms of the Orlik-Solomon algebra. There are more explicit combinatorial constructions that allow us to compute the components of the resonance variety, such as neighbourly partitions [23] and multinets [24]. Positive-dimensional components of the characteristic variety (also translated ones) have been described in terms of morphisms of algebraic varieties [19, Theorem 6.7], but zero-dimensional translated components are still mysterious.

One of the main problems that we found while collecting information about the translated components of the characteristic varieties is the lack of examples. Suciu [45, 46] and Cohen [7] provide some of them, but they are too few to try to induce a general behaviour. Characteristic varieties can be realized as zero locus of the ideal of the minors of the Alexander matrix (Theorem 3.6). Alternatively, they can be computed directly using an algebraic complex defined by Salvetti and Settepanella [43]. In both cases, the algorithms involve the computation of a huge amount of minors, which becomes unfeasible when the number of hyperplanes exceeds 10.

After the definition of a new, better algorithm, we managed to compute characteristic varieties of line arrangements that have not appeared in literature as far as we know. We collected the results in a catalogue, to which the whole Chapter 5 is dedicated. It appears that, at least in the arrangements that we consider, zero-dimensional translated components are not so rare. During the analysis of the results, we found something unexpected: *we found a way that allows us to obtain the translated components directly from the combinatorial data in almost all cases*. We don't mean that we have a combinatorial description of the translated components: *if* we know that such a component exists, *then* we are able to recover it also in a combinatorial way (except in one case).

Our combinatorial method relies on the notion of neighbourly partitions, which we already recalled when talking about the resonance variety. In particular, if π is a neighbourly partition, a homogeneous component of the characteristic variety can be recovered from the zero locus of the ideal

$$\mathcal{I}(\pi) \coloneqq \left(\prod_{j=1}^{n+1} T_j - 1 \right) + \left(\prod_{j \in P} T_j - 1 \;\middle|\; P \in \mathcal{P} \right)$$

where $\mathcal{P}$ is the set of the singular points not contained in a single block of π. Now, it is known that a homogeneous component has dimension at least 2, so partitions π such that $\dim \mathcal{I}(\pi) \leqslant 1$ have no use in the description of the homogeneous part of $\mathcal{V}(\mathcal{A})$. However, the definition of $\mathcal{I}(\pi)$ does *not* require that π is neighbourly: what happens then if we compute $\mathcal{I}(\pi)$ for other partitions?

Proposition. *For all arrangements $\mathcal{A}$ in our catalogue except $\mathcal{A}(11,1)$, if $\pi_{\mathcal{A}}$ is the double points partition of $\mathcal{A}$, the essential translated components of $\mathcal{V}(\mathcal{A})$ (if exist) appear as the zero locus of one ideal of the primary decomposition of (the radical of) $\mathcal{I}(\pi_{\mathcal{A}})$. For $\mathcal{A}(11,1)$ the two essential translated points do belong to the zero locus of $\mathcal{I}(\pi)$, but only as embedded components.*

All examples that we computed seem to lead to the conclusion that the translated part of the characteristic variety is combinatorial, but a lot of work has still to be done.

This strange behaviour of the translated components with respect to the neighbourly partition wouldn't have been possible without an improvement of the algorithm that computes the characteristic variety. We managed to write a version of the algorithm that does not require the computation of all the minors: this new algorithm tries to reduce the matrix in echelon form in order to compute its rank. The problem, of course, is that the coefficients are polynomials, so they can't be used to reduce the matrix unless we assume that they are different than zero. Therefore, whenever the algorithm tries to use a polynomial, it splits its computation tree, opening two new branches: in one of them, the polynomial is assumed to be zero, while in the other it is assumed to be different than zero. The algorithm then continues its computation on the two branches independently, and in the end it collects the results of all branches. Despite the large number of bifurcations, this new algorithm is far more efficient than the one we used previously, reducing computation time from weeks to minutes. We believe that this algorithm will help other researchers to study new examples, therefore we decided to include the actual code here. It is written in the `SageMath` language [40], a modern and powerful computer algebra system that itself contains already some procedures to study hyperplane arrangements.

In the second part of this work we concentrate on the theory of toric arrangements. These are defined in a way similar to the hyperplane arrangements, except that the ambient space is an algebraic torus instead of an affine or projective space and the hyperplanes are replaced by (possibly translated) subtori. Toric arrangements have been studied since the early 1990s, and over the last two decades several aspects have been investigated, both from combinatorial and topological points of view. In particular, as far as the topology of the complement is concerned, De Concini and Procesi determined the generators of the cohomology modules over $\mathbb{C}$ in the divisorial case, as well as the ring structure in the case of totally unimodular arrangements [14]; d'Antonio and Delucchi provided a presentation of the fundamental group for the complement of a divisorial complexified arrangement [18]; Callegaro, Delucchi and Pagaria computed the cohomology ring with integer coefficients [6, 38].

The problem of studying wonderful models for toric arrangement was first addressed by Moci in [36], where he described a construction of a non-projective model. Wonderful models were introduced by De Concini and Procesi in [13, 15], where they provided both a projective and a non-projective version of their construction. A *wonderful model* for the complement of an arrangement $\mathcal{M}(\mathcal{A})$ is a smooth variety $Y_{\mathcal{A}}$ containing $\mathcal{M}(\mathcal{A})$ as an open set and such that $Y_{\mathcal{A}} \setminus \mathcal{M}(\mathcal{A})$ is a divisor with normal crossings and smooth irreducible components.

In a recent article [12], De Concini and Gaiffi show how to construct a *projective wonderful model* for the complement of a toric arrangement $\mathcal{A}$. The key ingredient in this construction is a toric variety $X_{\mathcal{A}}$ with some good properties. This variety is obtained by subdividing a given fan in a suitable way. De Concini and Gaiffi provide an algorithm to do so, and we decided to implement it in the `SageMath` language in order to produce some meaningful examples.

Basing on the results of [12], in [11] the same authors describe a presentation of the cohomology ring of the wonderful model $Y_{\mathcal{A}}$ with integer coefficients; more precisely, they show that $H^*(Y_{\mathcal{A}}; \mathbb{Z})$ is isomorphic to a quotient of a polynomial ring with coefficients in $H^*(X_{\mathcal{A}}; \mathbb{Z})$. Starting from their work, we develop an algorithm that is able to compute the ideal of the relations involved in the presentation of $H^*(Y_{\mathcal{A}}; \mathbb{Z})$ as the quotient of a polynomial ring with $\mathbb{Z}$ coefficients. Actually, the wonderful model $Y_{\mathcal{A}}$ depends on the choice of a *building set* and our algorithm is able to compute the minimal one, which is the most desirable case from both the topological and the computational points of view. Unfortunately, we can't deal with toric arrangements in full generality, because we adopt an algorithm ([31]) based on a definition of toric arrangements that is different from the one we give. We hope to overcome this difficulty in the near future.

This work is structured as follows. In the first chapter we introduce the definition of hyperplane arrangements, as well as the basic constructions related to them. We focus on the idea of combinatorial properties, and we show that the Orlik-Solomon algebra is combinatorial, while the fundamental group is not.

The second chapter introduces the Milnor fibre, and describes briefly its topology and combinatorics. The main unanswered question regarding the Milnor fibre is a combinatoric description of its (co)homology. The study of this (co)homology leads to the notion of local systems on an arrangement, which we remember here. We recall also the algebraic complex that is able to compute the (co)homology of the complement of an arrangement with coefficients in a local system [26, 43].

Characteristic varieties are the main topic of the third chapter. After an introductory section, we present the classical methods to compute the characteristic variety of an arrangement. We then recall the definition of the resonance varieties of an arrangement and explore their connection with the characteristic varieties. Then we proceed to illustrate what is known in the literature about the combinatorics of resonance and characteristic varieties.

Our algorithms allow us to produce more examples of characteristic varieties of arrangements that have not appeared in the literature, as far as we know. Basing on those, we elaborated a couple of conjectures which we discovered to be false while writing this work. Nonetheless we decided to include the reasoning behind them, hoping that it will be useful in the development of new ideas.

The next two chapters have a more technical nature. The fourth chapter includes a detailed description of our main algorithms, together with the actual `SageMath` code. We decided to show algorithms for the computation of the algebraic complex of Chapter 2, for the computation of the first characteristic variety (both the classical and the improved one), and for the analysis of the neighbourly partitions. In addition to them, we define a class that helps us to manage arrangements which are difficult for a computer to deal with.

As we mentioned before, the fifth chapter is a catalogue of some projective line arrangements that we consider noteworthy. Some of them have already appeared in literature, while others are new. For each of them, we report the list of neighbourly partitions and the characteristic variety. To make order in the literature (almost each author uses his/her own nomenclature), we decided to report also the names used in other works to denote the same arrangement, and pictures of the most common forms in which it shows up.

The final two chapters are dedicated to the toric arrangements. In the sixth chapter, we recall the basic definitions regarding toric varieties and toric arrangements and we show two algorithms that can be used to subdivide a fan in order to get a good toric variety for an arrangement $\mathcal{A}$: the first is the one outlined in [12], while the other seems to output better fans, but works only in the 2-dimensional case. We produce some examples, explaining the behaviour of the two algorithms and underlining similarities and differences.

The last chapter is focused on the cohomology ring of a projective wonderful model for the complement of an arrangement. After a brief summary of the definitions and results that are needed to find the relations of the presentation of $H^*(Y_{\mathcal{A}}; \mathbb{Z})$, we delineate the algorithm and present some first examples of cohomology rings computed by it.

Chapter 1

Hyperplane Arrangements

In this first chapter we recall the basic definitions and constructions in the theory of hyperplane arrangements. This both serves as a general introduction to the topic and allows us to fix some notation.

In particular, the idea of *combinatorial properties* of an arrangement is defined, together with two important examples of objects associated with an arrangement, one which is combinatorial and one which is not.

1.1 Basic Definitions

First of all we introduce our main object of study.

Definition 1.1. Let V be a vector space over a field $\mathbb{K}$. An *(affine) hyperplane arrangement* $\mathcal{A}$ is a set of affine hyperplanes of V.

Definition 1.2. Let P be a projective space over a field $\mathbb{K}$. A *projective hyperplane arrangement* $\mathcal{A}$ is a set of projective hyperplanes of P.

Unless otherwise specified, we will assume that all the arrangements are finite, and all the spaces are finite-dimensional.

Definition 1.3. The *dimension* $\dim(\mathcal{A})$ of a hyperplane arrangement $\mathcal{A}$ is the dimension of the underlying space. For an affine arrangement, the *rank* $\mathrm{rk}(\mathcal{A})$ is the dimension of the subspace spanned by the normal vectors of the hyperplanes of $\mathcal{A}$. An affine arrangement $\mathcal{A}$ is called *essential* if $\dim(\mathcal{A}) = \mathrm{rk}(\mathcal{A})$.

An affine hyperplane $H \subseteq V$ is given by

$$H = \{\boldsymbol{v} \in V \mid \alpha(\boldsymbol{v}) = a\}$$

for some non-degenerate linear form $\alpha\colon V \to \mathbb{K}$ and some constant $a \in \mathbb{K}$. Notice that, if (α, a) is a pair that defines an hyperplane H as above, then also $(k\alpha, ka)$ defines the same hyperplane for every $k \in \mathbb{K}^*$.

Definition 1.4. Let $\mathcal{A}$ be an arrangement in an m-dimensional vector space V and, for each $H \in \mathcal{A}$, let (α_H, a_H) be a pair that defines H. The polynomial

$$Q_{\mathcal{A}}(\mathbf{X}) \coloneqq \prod_{H \in \mathcal{A}} (\alpha_H(\mathbf{X}) - a_H) \in \mathbb{K}[\mathbf{X}] = \mathbb{K}[X_1, \dots, X_m]$$

is called *defining polynomial* of the arrangement, and it is defined up to a non-zero constant.

Remark. The same definition applies to projective arrangements in a projective space P; in this case a hyperplane is given by

$$H = \{\mathbf{p} \in P \mid \alpha(\mathbf{p}) = 0\}$$

where $\alpha \colon P \to \mathbb{K}$ is a homogeneous polynomial of degree one.

Notation 1.5. If $\mathcal{A}$ is an arrangement of the space V, and $\mathcal{B} \subseteq \mathcal{A}$, then

$$\cap \mathcal{B} \coloneqq \bigcap_{H \in \mathcal{B}} H \quad \text{and} \quad \cup \mathcal{B} \coloneqq \bigcup_{H \in \mathcal{B}} H.$$

By convention, $\cap \varnothing = V$.

Definition 1.6. The *complement* of a hyperplane arrangement is the set

$$\mathcal{M}(\mathcal{A}) \coloneqq V \setminus (\cup \mathcal{A}) = V \setminus \bigcup_{H \in \mathcal{A}} H.$$

Remark. Obviously $\cup \mathcal{A} = \{\boldsymbol{v} \in V \mid Q_{\mathcal{A}}(\boldsymbol{v}) = 0\}$ and $\mathcal{M}(\mathcal{A}) = \{\boldsymbol{v} \in V \mid Q_{\mathcal{A}}(\boldsymbol{v}) \neq 0\}$.

The space $\mathcal{M}(\mathcal{A})$ is probably the most studied topological object in the theory of hyperplane arrangements. Its characteristics depend deeply on the field over which the vector space V is defined. For example, if $V = \mathbb{R}^m$, the topology is trivial: $\mathcal{M}(\mathcal{A})$ is the union of a certain number of connected components, and all of them are convex. It is interesting in this case to *count* the components, and quite surprisingly there is a univariate polynomial (the *characteristic polynomial* of the arrangement) that encodes this information, among a lot of other properties. A further analysis of this polynomial is beyond the scope of this work.

Example 1.1 (Braid arrangement). Consider the vector space $\mathbb{K}^m$ with coordinates $(v_1, \dots, v_m)$ and let $H_{ij} \subseteq \mathbb{K}^m$ be the hyperplane

$$H_{ij} \coloneqq \{\boldsymbol{v} \in \mathbb{K}^m \mid v_i = v_j\}.$$

The set

$$\mathcal{B}r_m \coloneqq \{H_{ij} \mid 1 \leqslant i < j \leqslant m\}$$

is the *braid arrangement* of $\mathbb{K}^m$. Its cardinality is

$$\#(\mathcal{B}r_m) = \binom{m}{2}.$$

The arrangement is *not* essential, because $\langle(1,\dots,1)\rangle \subseteq H_{ij}$ for all i and j. By intersecting each hyperplane with any subspace W such that $\langle(1,\dots,1)\rangle \oplus W = \mathbb{K}^m$, we obtain an $(m-1)$-dimensional essential arrangement, again called braid arrangement $\mathcal{B}r_m$. The context makes clear if $\mathcal{B}r_m$ denotes the non-essential m-dimensional arrangement or this last one.

Definition 1.7. An affine hyperplane arrangement $\mathcal{A}$ is *central* if $\cap\mathcal{A} \neq \varnothing$. After a suitable change of coordinates, it can be assumed that $0 \in \cap\mathcal{A}$—in this case the arrangement is also called *linear*, and the defining polynomial is homogeneous.

If $\mathcal{A}$ is an arrangement of lines in the (affine or projective) plane, we give some additional definitions.

Definition 1.8. The *singular points* $\mathcal{S}ing(\mathcal{A})$ of a line arrangement $\mathcal{A}$ are the intersection points of the lines of $\mathcal{A}$:

$$\mathcal{S}ing(\mathcal{A}) := \{\ell \cap \ell' \mid \ell, \ell' \in \mathcal{A},\ \ell \neq \ell'\}.$$

If $P \in \mathcal{S}ing(\mathcal{A})$, its *multiplicity* $m(P)$ is the number of lines passing through it:

$$m(P) := \#(\{\ell \in \mathcal{A} \mid P \in \ell\}).$$

Points P with $m(P) = 2, 3, 4, \dots$ are called *double, triple, quadruple... points*.

1.2 Coning and Deconing

Let $\mathcal{A}$ be a central arrangement in $V = \mathbb{K}^m$. Then a hyperplane $H \in \mathcal{A}$ defines a projective hyperplane $\overline{H} \subseteq \mathbb{P}(V)$. The family of projective hyperplanes

$$\overline{\mathcal{A}} := \{\overline{H} \mid H \in \mathcal{A}\} \subseteq \mathbb{P}(V)$$

is a projective hyperplane arrangement, called *projectivization* of $\mathcal{A}$. Obviously $Q_{\mathcal{A}} = Q_{\overline{\mathcal{A}}}$. On the other hand, let $\mathcal{A}$ be an arrangement in $\mathbb{P}(V)$; a hyperplane $H \in \mathcal{A}$ defines a hyperplane $\widehat{H} \subseteq V$ passing through the origin. The family of linear hyperplanes

$$\widehat{\mathcal{A}} := \{\widehat{H} \mid H \in \mathcal{A}\} \subseteq V$$

is a linear hyperplane arrangement, called *deprojectivization* of $\mathcal{A}$. Also in this case $Q_{\mathcal{A}} = Q_{\widehat{\mathcal{A}}}$.

Now let $\mathcal{A}$ be an affine (not necessarily linear) arrangement of r hyperplanes in $V = \mathbb{K}^m$, and for each $H \in \mathcal{A}$ let (α_H, a_H) be a pair that defines H. The arrangement of $r+1$ hyperplanes in $V \times \mathbb{K} = \mathbb{K}^{m+1}$ given by

$$\{\{(\boldsymbol{v}, y) \in V \times \mathbb{K} \mid \alpha_H(\boldsymbol{v}) = a_H y\} \mid H \in \mathcal{A}\} \cup \{\{(\boldsymbol{v}, y) \in V \times \mathbb{K} \mid y = 0\}\},$$

where $\boldsymbol{v}$ are the variables for V and y is the new variable for $\mathbb{K}$, is called *coning* of $\mathcal{A}$ and denoted by $c\mathcal{A}$. The defining polynomials of $\mathcal{A}$ and $c\mathcal{A}$ satisfy

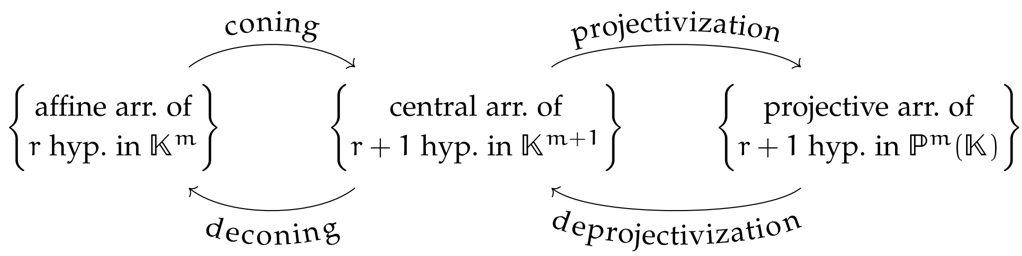
$$Q_{c\mathcal{A}}(X_1, \dots, X_m, Y) = Y^{m+1} \cdot Q_{\mathcal{A}}(X_1/Y, \dots, X_m/Y).$$

The opposite operation is defined easily. Let $\mathcal{A} = \{H_1, \dots, H_{r+1}\}$ be a linear arrangement of $r+1$ hyperplanes in $\mathbb{K}^{m+1}$, and choose coordinates such that $H_{r+1} = \{\boldsymbol{x} \in \mathbb{K}^{m+1} \mid x_{m+1} = 0\}$. Identify the hyperplane $K := \{\boldsymbol{x} \in \mathbb{K}^{m+1} \mid x_{m+1} = 1\}$ with $\mathbb{K}^m$. The arrangement

$$\{H_i \cap K \mid i = 1, \dots, r\}$$

of r hyperplanes in $\mathbb{K}^m$ is called *deconing* of $\mathcal{A}$ and denoted by $d\mathcal{A}$; its defining polynomial is

$$Q_{d\mathcal{A}}(X_1, \dots, X_m) = Q_{\mathcal{A}}(X_1, \dots, X_m, 1).$$



Once we have defined the two pairs of operations "projectivization/deprojectivization" and "coning/deconing", it is easy to see what happens if we combine them. It turns out that, if $\mathcal{A}$ is an affine arrangement of r hyperplanes in $\mathbb{K}^m$, $\overline{c\mathcal{A}}$ is the projective arrangement of $r+1$ hyperplanes in $\mathbb{P}^m(\mathbb{K})$ obtained by adding the hyperplane at infinity to $\mathcal{A}$. In particular,

$$\mathbb{P}^m(\mathbb{K}) \setminus (\cup \overline{c\mathcal{A}}) = V \setminus (\cup \mathcal{A})$$

where $V = \mathbb{K}^m$ on the right side is identified with the affine chart $\{y = 0\}$ in $\mathbb{P}^m(\mathbb{K}) = \mathbb{P}(\mathbb{K}^m \times \mathbb{K})$.

Naturally, if we begin with a projective arrangement $\mathcal{A}$ of $r+1$ hyperplanes in $\mathbb{P}^m(\mathbb{K})$, computing the affine arrangement $d\widehat{\mathcal{A}}$ corresponds to sending to infinity the hyperplane chosen for the deconing and looking at $\mathcal{A}$ in the relative affine chart. We still have

$$V \setminus (\cup d\widehat{\mathcal{A}}) = \mathbb{P}^m(\mathbb{K}) \setminus (\cup \mathcal{A}).$$

Notation 1.9. We will use $p\mathcal{A}$ and $a\mathcal{A}$ as abbreviations respectively of $\overline{c\mathcal{A}}$ and $d\widehat{\mathcal{A}}$.

1.3 Combinatorial Properties

The combinatorial data associated to a hyperplane arrangement is encoded in the so-called *intersection poset*.

Definition 1.10. The *intersection poset* of an arrangement $\mathcal{A}$ is the set

$$L(\mathcal{A}) := \{\cap \mathcal{B} \mid \mathcal{B} \subseteq \mathcal{A},\ \cap \mathcal{B} \neq \varnothing\}$$

partially ordered by reverse inclusion. Note that $V \in L(\mathcal{A})$, because $V = \cap \varnothing$.

Remark. If $\mathcal{A}$ is central, the poset $L(\mathcal{A})$ is actually a lattice.

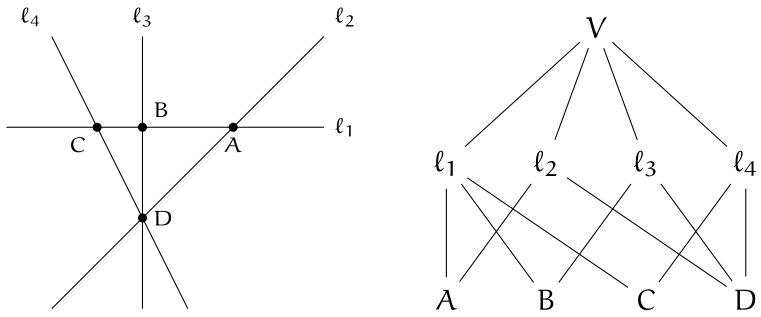


Figure 1.1: An example of intersection poset of an arrangement of four lines in $\mathbb{R}^2$. On the right it is shown the Hasse diagram of $L(\mathcal{A})$.

Definition 1.11. We say that an object $P(\mathcal{A})$ associated to an arrangement $\mathcal{A}$ is *combinatorial* (or *combinatorially determined*) if it depends only on $L(\mathcal{A})$, that is to say, if for any two arrangements $\mathcal{A}_1$, $\mathcal{A}_2$, $L(\mathcal{A}_1) \simeq L(\mathcal{A}_2)$ implies $P(\mathcal{A}_1) \simeq P(\mathcal{A}_2)$. Similarly, a property $p(\mathcal{A})$ is combinatorial if $L(\mathcal{A}_1) \simeq L(\mathcal{A}_2)$ implies $p(\mathcal{A}_1) \Leftrightarrow p(\mathcal{A}_2)$.

It is not clear what is combinatorial among the many interesting properties of an arrangement. Moreover, there are meaningful examples of both combinatorial objects and non-combinatorial ones.

Proposition 1.12. *The cohomology ring* $H^*(\mathcal{M}(\mathcal{A}); \mathbb{Z})$ *is combinatorial.*

Proposition 1.13. *The fundamental group* $\pi_1(\mathcal{M}(\mathcal{A}))$ *is* not *combinatorial.*

The cohomology ring will be studied in Section 1.4. We show here a famous example, due to Rybnikov, of two arrangements which have isomorphic intersection posets but non-isomorphic fundamental groups.

Example 1.2 (Rybnikov's Example [2, 39]). Consider the following lines in $\mathbb{P}^2(\mathbb{C})$ (with coordinates $[x : y : z]$):

$$\begin{gathered}\ell_0 = \{x = 0\}, \quad \ell_1 = \{y = 0\}, \quad \ell_2 = \{x = y\}, \quad \ell_3 = \{z = 0\}, \quad \ell_4 = \{x = z\},\\ \ell_5^+ = \{x + \omega y = 0\}, \quad \ell_6^+ = \{z + \omega y = (\omega + 1)x\}, \quad \ell_7^+ = \{(\omega + 1)y + z = x\},\\ \ell_5^- = \{x + \overline{\omega} y = 0\}, \quad \ell_6^- = \{z + \overline{\omega} y = (\overline{\omega} + 1)x\}, \quad \ell_7^- = \{(\overline{\omega} + 1)y + z = x\}\end{gathered}$$

where $\omega = e^{2\pi i/3}$. Let

$$\begin{aligned} L_\omega &= \{\ell_0, \ell_1, \ell_2, \ell_3, \ell_4, \ell_5^+, \ell_6^+, \ell_7^+\},\\ L_{\overline{\omega}} &= \{\ell_0, \ell_1, \ell_2, \ell_3, \ell_4, \ell_5^-, \ell_6^-, \ell_7^-\}\end{aligned}$$

and let $\rho_\omega, \rho_{\overline{\omega}} \colon \mathbb{P}^2(\mathbb{C}) \to \mathbb{P}^2(\mathbb{C})$ projective transformations such that

- $\rho_\omega(\ell_i) = \ell_i$ and $\rho_{\overline{\omega}}(\ell_i) = \ell_i$, for $i = 0, 1, 2$;
- the lines $\{\rho_\omega(\ell) \mid \ell \in L_\omega\}$ intersect the lines of L_ω only in double points outside ℓ_0, ℓ_1 and ℓ_2;
- the lines $\{\rho_{\overline{\omega}}(\ell) \mid \ell \in L_{\overline{\omega}}\}$ intersect the lines of L_ω only in double points outside ℓ_0, ℓ_1 and ℓ_2.

Finally, let[1]

$$\begin{aligned} \mathcal{A}_\omega &:= L_\omega \cup \{\rho_\omega(\ell) \mid \ell \in L_\omega\},\\ \mathcal{A}_{\overline{\omega}} &:= L_{\overline{\omega}} \cup \{\rho_{\overline{\omega}}(\ell) \mid \ell \in L_\omega\}.\end{aligned}$$

The two arrangements have isomorphic intersection posets (their Hasse diagram is pictured in Figure 1.2), but the fundamental groups of the complements are not isomorphic. In fact, Rybnikov [39] explicitly builds an invariant of a group G (that depends only on the lower central series of G itself) that is able to distinguish between $\pi_1(\mathcal{M}(\mathcal{A}_\omega))$ and $\pi_1(\mathcal{M}(\mathcal{A}_{\overline{\omega}}))$. A more detailed description of this invariant is beyond the scope of this work.

1.4 The Orlik-Solomon Algebra

In order to prove that the cohomology ring of the complement of an arrangement $\mathcal{A}$ is combinatorial, we show that, using the data of $L(\mathcal{A})$, it is possible to define an algebra $A(\mathcal{A})$ that is isomorphic (as a graded algebra) to the cohomology ring itself.

Suppose that $\mathcal{A}$ is a central arrangement in a vector space V. For each hyperplane $H \in \mathcal{A}$ define a symbol e_H. If $\mathcal{A} = \{H_1, \dots, H_r\}$, let e_i be an abbreviation for e_{H_i}.

[1] This definition depends on the choice of ρ_ω and $\rho_{\overline{\omega}}$. However, it is easy to verify that both the intersection poset and the fundamental group of $\mathcal{A}_\omega$ do not depend on this choice, nor do the ones of $\mathcal{A}_{\overline{\omega}}$.

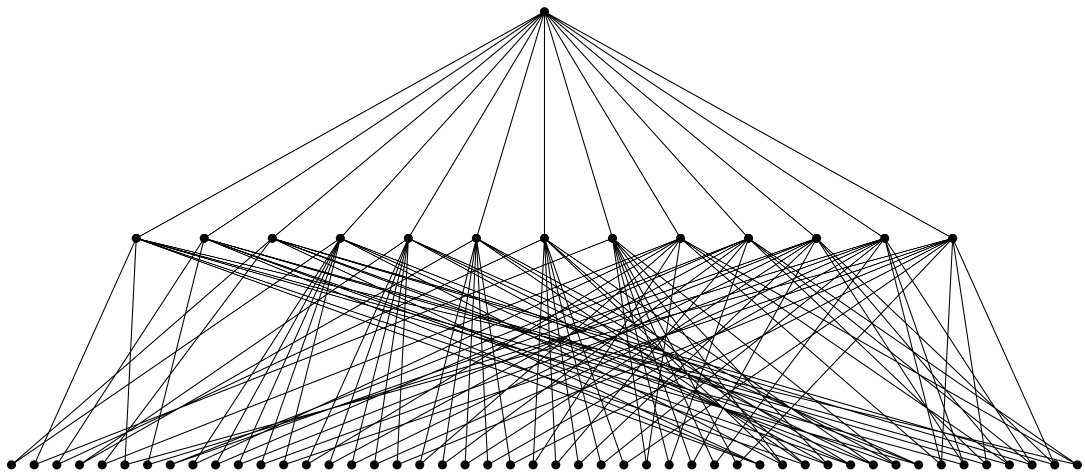


Figure 1.2: Hasse diagram of the (isomorphism class of the) intersection posets of $\mathcal{A}_\omega$ and $\mathcal{A}_{\overline{\omega}}$.

Let R be a commutative ring with unit and consider the free R-module generated by $\{e_i \mid i = 1, \dots, r\}$:

$$E_1 \coloneqq \langle e_i \mid i = 1, \dots, r \rangle.$$

Let $E \coloneqq \Lambda E_1$ be the exterior algebra of E_1. Recall that E is graded as

$$E = \bigoplus_{p=0}^{r} E_p$$

where $E_0 = R$ and E_p is generated over R by all products $e_{i_1} \wedge \cdots \wedge e_{i_p}$ with $(H_{i_1}, \dots, H_{i_p})$ p-tuple of hyperplanes of $\mathcal{A}$. Define a R-linear map $\partial \colon E \to E$ such that $\partial(1) = 0$, $\partial(e_i) = 1$ for all $i = 1, \dots, r$ and, if $p \geqslant 2$,

$$\partial(e_{i_1} \wedge \cdots \wedge e_{i_p}) = \sum_{k=1}^{p} (-1)^{k-1} e_{i_1} \wedge \cdots \wedge \widehat{e_{i_k}} \wedge \cdots \wedge e_{i_p}$$

where $\widehat{e_{i_k}}$ means that e_{i_k} is missing from the product.

Notation 1.14. If $S = (H_{i_1}, \dots, H_{i_p})$ is a p-tuple of hyperplanes, denote

$$|S| \coloneqq p, \quad \cap S \coloneqq H_{i_1} \cap \cdots \cap H_{i_p}, \quad e_S \coloneqq e_{i_1} \wedge \cdots \wedge e_{i_p}.$$

Since $\mathcal{A}$ is central, $\cap S \in L(\mathcal{A})$ for all S. Moreover, for $p = 0$, $S = ()$ and put $e_S = 1$ and $\cap S = V$. It is clear that $\operatorname{codim}(\cap S) \leqslant |S|$.

Definition 1.15. A p-tuple S is *independent* if $\operatorname{codim}(\cap S) = |S|$, and *dependent* if $\operatorname{codim}(\cap S) < |S|$.

Let $I \subseteq E$ be the ideal generated by $\{\partial(e_S) \mid S \text{ is dependent}\}$. I is generated by homogeneous elements, therefore it is a graded ideal:

$$I = \bigoplus_{p=0}^{r} I_p$$

with $I_p = E_p \cap I$.

Definition 1.16. The *Orlik-Solomon algebra* $A(\mathcal{A})$ of an arrangement $\mathcal{A}$ is the quotient

$$A(\mathcal{A}) := {}^{E}/_{I}.$$

Let $\varphi\colon E \to A$ be the projection. It is not difficult to show that A is graded

$$A = \bigoplus_{p=0}^{r} A_p$$

with $A_p = \varphi(E_p)$. Moreover $A_0 = R$ and $A_1 = \langle a_H \mid H \in \mathcal{A} \rangle$ where $a_H = \varphi(e_H)$.

If $\mathcal{A}$ is not central, the construction differs slightly. The problem is that for a p-tuple S the intersection $\cap S$ may be empty; the definition of dependent p-tuple becomes the following.

Definition 1.15*. A p-tuple S is *dependent* if $\cap S \neq \varnothing$ and $\operatorname{codim}(\cap S) < |S|$.

The ideal I is now generated by

$$\{e_S \mid \cap S = \varnothing\} \cup \{\partial(e_S) \mid S \text{ is dependent}\}$$

and the Orlik-Solomon algebra is defined as $A(\mathcal{A}) = {}^{E}/_{I}$.

Now that we showed the construction of the Orlik-Solomon algebra of an arrangement $\mathcal{A}$, we may state the main result.

Theorem 1.17 (see also Theorem 5.90 of [37])**.** *Let $\mathcal{A}$ be a hyperplane arrangement in a complex vector space, and choose $R = \mathbb{Z}$. The Orlik-Solomon algebra $A(\mathcal{A})$ and the integer cohomology ring $H^*(\mathcal{M}(\mathcal{A}); \mathbb{Z})$ are isomorphic as graded $\mathbb{Z}$-algebras.*

The proof of this theorem requires some more effort, so it won't be reported here. The important fact is that a topological invariant associated with an arrangement actually depends on its combinatorics; the next natural question is what other topological invariants are combinatorial. In the next chapter we will focus our attention on one particular invariant: the so-called Milnor fibre.

Chapter 2

Milnor Fibre and Local Systems

In this chapter we deal with the Milnor fibre, which is a smooth manifold associated with a central complex arrangement. One of the main lines of research is to investigate the topological properties of this object, and try to understand what of them may be combinatorial. In particular, we focus on the first homology group of the Milnor fibre, which is isomorphic to the first homology group of the complement of the arrangement with local coefficients (see Theorem 2.6).

Local systems themselves are described in Section 2.2. The main task is to compute homology with coefficients in a local system, and in Section 2.3 we recall the construction of an algebraic complex that is able to do so.

2.1 The Milnor Fibre and Its Monodromy

We begin with the definition of the Milnor fibre in a general setting. Let $f\colon U \to \mathbb{C}$ be an analytic function of m complex variables, with $U \subseteq \mathbb{C}^m$ neighbourhood of 0, and suppose that $f(0) = 0$. Let $Z \coloneqq \{z \in U \mid f(z) = 0\}$, $S_\varepsilon \coloneqq \{z \in \mathbb{C}^m \mid |z| = \varepsilon\}$, and $K_\varepsilon \coloneqq Z \cap S_\varepsilon$. Then the map

$$\begin{array}{rccc} \varphi\colon & S_\varepsilon \setminus K_\varepsilon & \longrightarrow & S^1 \\ & z & \longmapsto & f(z)/|f(z)| \end{array}$$

is well-defined.

Theorem 2.1 (Fibration [35]). *There exists $\varepsilon_0 > 0$ such that for every $\varepsilon \leqslant \varepsilon_0$ the space $S_\varepsilon \setminus K_\varepsilon$ is a smooth fibre bundle over S^1, with projection mapping φ.*

Therefore each fibre $G_\theta \coloneqq \varphi^{-1}(e^{i\theta})$ is a smooth manifold of (real) dimension $2(m-1)$.

Definition 2.2. The *Milnor fibre* of f is the fibre $G_0 = \varphi^{-1}(1)$.

Definition 2.3. Let $h_t \colon G_0 \to G_t$ a one-parameter family of homeomorphisms, where $t \in [0, 2\pi]$, such that $h_0 = \mathrm{Id}$. The homeomorphism $h := h_{2\pi}$ is the *characteristic homeomorphism* of the Milnor fibre.

Remark. The homeomorphism h depends on the choice of the family h_t, but its homotopy class is uniquely determined.

If f is a polynomial, we may further characterise the Milnor fibre of f.

Proposition 2.4. *Let $f(z_1, \dots, z_m)$ be a homogeneous polynomial of degree k. The Milnor fibre of f is diffeomorphic to the nonsingular hypersurface*

$$F := \{z \in \mathbb{C}^m \mid f(z) = 1\}$$

and we may choose as characteristic homeomorphism the map

$$\begin{array}{rccc} h\colon & F & \longrightarrow & F \\ & z & \longmapsto & e^{2\pi i/k} z. \end{array}$$

Now we apply this general setting to the case of hyperplane arrangements. Let $\mathcal{A}$ be a central arrangement of $n+1$ hyperplanes in the complex vector space $\mathbb{C}^m$. Recall that the defining polynomial $Q_{\mathcal{A}} \in \mathbb{C}[Z_1, \dots, Z_m]$ is homogeneous of degree $n+1$; consider it as a map

$$Q_{\mathcal{A}} \colon \mathbb{C}^m \to \mathbb{C}.$$

Definition 2.5. The restriction

$$Q_{\mathcal{A}}\big|_{\mathcal{M}(\mathcal{A})} \colon \mathcal{M}(\mathcal{A}) \to \mathbb{C}^*$$

defines a fibration, whose fibre $F_{\mathcal{A}} := Q_{\mathcal{A}}^{-1}(1)$ is called *Milnor fibre* of the arrangement. The map

$$\begin{array}{rccc} h\colon & F_{\mathcal{A}} & \longrightarrow & F_{\mathcal{A}} \\ & z & \longmapsto & \lambda z \end{array}$$

with $\lambda = e^{2\pi i/(n+1)}$ is an automorphism of $F_{\mathcal{A}}$, called *geometric monodromy* of the Milnor fibre.

There are some open problems regarding the Milnor fibre of an arrangement, for example it is not known in general whether its Betti numbers are combinatorial (not even for the first Betti number in the case $m = 3$).

For our purpose it is sufficient to consider just the first homology group $H_1(F_{\mathcal{A}}; \mathbb{C})$. The geometric monodromy induces a linear map

$$h_1 \colon H_1(F_{\mathcal{A}}; \mathbb{C}) \to H_1(F_{\mathcal{A}}; \mathbb{C}).$$

The following result is well-known (see also [9]).

Theorem 2.6. *There is a* $\mathbb{C}[T^{\pm 1}]$*-module isomorphism*

$$H_1(F_{\mathcal{A}};\mathbb{C}) \simeq H_1(\mathcal{M}(\mathcal{A});\mathbb{C}[T^{\pm 1}])$$

under which the monodromy action corresponds to the multiplication by T*, i.e.* $T \cdot [a] = h_1([a])$ *for* $[a] \in H_1(F_{\mathcal{A}};\mathbb{C})$.

The ring $\mathbb{C}[T^{\pm 1}]$ is a principal ideal domain, so $H_1(\mathcal{M}(\mathcal{A});\mathbb{C}[T^{\pm 1}])$ decomposes into cyclic modules. Moreover, the monodromy action has order that divides $n+1$, therefore the polynomial $T^{n+1}-1$ annihilates the homology. It follows that there is a decomposition

$$H_1(\mathcal{M}(\mathcal{A});\mathbb{C}[T^{\pm 1}]) \simeq \bigoplus_{d|n+1} \left(\mathbb{C}[T^{\pm 1}]/(\varphi_d)\right)^{b_d} \tag{2.1}$$

where φ_d is the d-th cyclotomic polynomial.

Definition 2.7. The central arrangement $\mathcal{A}$ of $n+1$ hyperplanes in $\mathbb{C}^m$ is *a-monodromic* if

$$H_1(\mathcal{M}(\mathcal{A});\mathbb{C}[T^{\pm 1}]) \simeq \mathbb{C}^n \left[\simeq \left(\mathbb{C}[T^{\pm 1}]/(T-1)\right)^n \right].$$

Remark. Since $h^{n+1} = \mathrm{Id}$, there is a decomposition into eigenspaces

$$H_1(F_{\mathcal{A}};\mathbb{C}) = \bigoplus_{\eta^{n+1}=1} H_1(F_{\mathcal{A}};\mathbb{C})_\eta.$$

It is clear that, under the isomorphism given by Theorem 2.6, if η is a d-th root of unity

$$H_1(F_{\mathcal{A}};\mathbb{C})_\eta \simeq \left(\mathbb{C}[T^{\pm 1}]/(\varphi_d)\right)^{b_d}.$$

An arrangement is a-monodromic if and only if in the decomposition (2.1) we have $b_1 = n$ and $b_d = 0$ for all $d \neq 1$. Therefore, as the name suggest, a-monodromicity is equivalent to triviality of the monodromy action.

How does the combinatorics of arrangements relate to Milnor fibres and a-monodromicity? A general answer is not known yet. There are some results, especially for central arrangements in $\mathbb{C}^3$ (i.e. projective line arrangements), see for example [3, 42, 47, 48]. Unfortunately these results lead to other questions, that are still in form of conjectures. Here we report just one of them, taken from [42].

Definition 2.8. Let $\mathcal{A} = \{H_1, \dots, H_{n+1}\}$ be a central arrangement in $\mathbb{C}^3$ and let $\overline{\mathcal{A}} = \{\overline{H}_1, \dots, \overline{H}_{n+1}\}$ be its projectivization in $\mathbb{P}^2(\mathbb{C})$. The *double points graph* $\Gamma(\mathcal{A})$ is the graph such that:

- the set of vertices is $\{\overline{H}_1, \dots, \overline{H}_{n+1}\}$;
- there is an edge $\{\overline{H}_i, \overline{H}_j\}$ if and only if $\overline{H}_i \cap \overline{H}_j$ is a double point (see Definition 1.8).

Conjecture 2.9 ([42]). *Let $\mathcal{A}$ be a central arrangement in $\mathbb{C}^3$. If $\Gamma(\mathcal{A})$ is connected, then $\mathcal{A}$ is a-monodromic.*

In [42], the authors state that "this conjecture is supported by several 'experiments', since all computations we made confirm it. Also, all non-trivial monodromy examples we know have disconnected graph Γ". However, as the number of hyperplanes increases, the volume of computation grows so much that computers can't provide more examples to better understand the situation.

2.2 Local Systems

In the previous section we saw that the homology of the Milnor fibre of an arrangement $\mathcal{A}$ is isomorphic to the homology of the complement $\mathcal{M}(\mathcal{A})$ with coefficients in the Laurent polynomial ring $\mathbb{C}[T^{\pm 1}]$, where the monodromy action corresponds to the multiplication by T. This is actually a particular case of *homology with local coefficients*, or *with coefficients in a local system*. Let us give a definition here.

Definition 2.10. Let $\mathcal{A}$ be an affine arrangement of n hyperplanes in the complex vector space $\mathbb{C}^m$ with complement $M = \mathcal{M}(\mathcal{A})$ and let R be a commutative ring with unity. A *(rank-1) local system* is a structure of $\pi_1(M)$-module on R.

In other words, a local system is a pair (R, ρ) where R is a commutative ring with unity and $\rho\colon \pi_1(M) \to \mathcal{A}ut(R)$ is a group homomorphism. It is usual to denote with R_ρ the ring R equipped with this $\pi_1(M)$-module structure.

Definition 2.11. A local system (R, ρ) is *abelian* if $\mathrm{Im}(\rho) \leqslant \mathcal{A}ut(R)$ is an abelian group.

For an abelian local system (R, ρ) the map $\rho\colon \pi_1(M) \to \mathcal{A}ut(R)$ actually factors through a map $\tilde{\rho}\colon H_1(M; \mathbb{Z}) \to \mathcal{A}ut(R)$:

$$\begin{array}{ccc} \pi_1(M) & \xrightarrow{\rho} & \mathcal{A}ut(R) \\ {\scriptstyle ab}\downarrow & \nearrow_{\tilde{\rho}} & \\ H_1(M; \mathbb{Z}) & & \end{array}$$

Recall that $H_1(M; \mathbb{Z})$ is a free abelian group of rank n generated by geometric loops $\beta_1, \ldots, \beta_n$ around the hyperplanes, so a map $\tilde{\rho}$ is defined once we know the images of the β_i's.

Remark. Local system can be defined in a more general situation. Let X be a path-connected space with universal covering $\tilde{X}$ and fundamental group $\pi = \pi_1(X)$. A local system for X is a (left) π-module L. Homology with coefficients in L can be defined in the following way: let $C_k(\tilde{X})$ be the group of (singular) k-chains in $\tilde{X}$; the action of π

on $\tilde{X}$ induces an action on $C_k(\tilde{X})$, making it a left π-module. Then it is well-defined the tensor product

$$C_k(X;L) \coloneqq C_k(\tilde{X}) \underset{\pi}{\otimes} L.$$

The map $\partial \otimes \mathrm{Id}$ turns the set $\{C_k(X;L)\}$ into a chain complex, whose homology $H_*(X;L)$ is by definition the *homology of* X *with coefficients in the local system* L. For cohomology, it is sufficient to consider the sets $C^k(X;L) = \mathcal{H}om_\pi(C_k(\tilde{X}), L)$ of π-module homomorphisms between $C_k(\tilde{X})$ and L; these groups form a cochain complex, whose cohomology $H^*(X;L)$ is by definition the *cohomology of* X *with coefficients in the local system* L.

We will now come back to our hyperplane arrangements and focus on $R = \mathbb{C}$. In this case, $\mathcal{A}ut(\mathbb{C}) \simeq \mathbb{C}^*$ is an abelian group itself, therefore in order to define a local system it is sufficient to choose a non-zero complex number t_i for every generator β_i of $H_1(M;\mathbb{Z})$. This brings to a 1-1 correspondence between rank-1 abelian local systems over $\mathbb{C}$ and the points of the complex torus $(\mathbb{C}^*)^n$, where a point $\mathbf{t} = (t_1, \dots, t_n) \in (\mathbb{C}^*)^n$ defines the local system $\mathbb{C}_\mathbf{t}$ given by the map

$$\begin{array}{ccc} H_1(M;\mathbb{Z}) & \longrightarrow & \mathbb{C}^* \\ \beta_i & \longmapsto & t_i. \end{array}$$

Another interesting case is $R = \mathbb{C}[T_1^{\pm 1}, \dots, T_n^{\pm 1}]$, the multivariate Laurent polynomial ring, with the map

$$\begin{array}{ccc} H_1(M;\mathbb{Z}) & \longrightarrow & \mathcal{A}ut(\mathbb{C}[T_1^{\pm 1}, \dots, T_n^{\pm 1}]) \\ \beta_i & \longmapsto & \big(p(\mathbf{T}) \mapsto T_i \cdot p(\mathbf{T})\big). \end{array}$$

The last example is $R = \mathbb{C}[T^{\pm 1}]$, the univariate Laurent polynomial ring, with the action

$$\begin{array}{ccc} H_1(M;\mathbb{Z}) & \longrightarrow & \mathcal{A}ut(\mathbb{C}[T^{\pm 1}]) \\ \beta_i & \longmapsto & \big(p(T) \mapsto T \cdot p(T)\big). \end{array}$$

If $\mathcal{A}$ is central, homology with coefficients in this local system is isomorphic to the homology of the Milnor fibre (see Theorem 2.6).

Remark. The last example allows us to define a-monodromicity for an affine (not necessarily central) arrangement. In fact, that local system is defined also when $\mathcal{A}$ is not central. Thus we say that *by definition* an affine arrangement is a-monodromic if

$$H_1(M;\mathbb{C}[T^{\pm 1}]) \simeq \left(\mathbb{C}[T^{\pm 1}]/(T-1)\right)^{n-1}.$$

Of course this definition agrees with Definition 2.7 for central arrangements. It is natural to ask how a-monodromicity behaves with respect to coning and deconing. It turns out (see [42]) that

- $\mathcal{A}$ is a-monodromic $\Rightarrow$ $c\mathcal{A}$ is a-monodromic;
- if $\mathcal{A}$ is central, $\mathcal{A}$ is a-monodromic $\not\Rightarrow$ $d\mathcal{A}$ is a-monodromic.

2.3 An Algebraic Complex

The next step is to compute the homology with local coefficients. In [43], the authors describe an algebraic complex that is able to compute the local coefficients homology for the complement of a *complexified real arrangement*, i.e. an arrangement in the complex space defined by real equations. More specifically, given a real arrangement $\mathcal{A}$ in $\mathbb{R}^m$, define

$$\mathcal{A}_{\mathbb{C}} \coloneqq \{\ell \otimes \mathbb{C} \mid \ell \in \mathcal{A}\} \subseteq \mathbb{C}^m \ (\simeq \mathbb{R}^m \underset{\mathbb{R}}{\otimes} \mathbb{C})$$

and consider the space $M \coloneqq \mathcal{M}(\mathcal{A}_{\mathbb{C}}) \subseteq \mathbb{C}^m$. It is known that M is homotopically equivalent to an explicit CW-complex $\mathcal{S}(\mathcal{A})$, the *Salvetti complex* ([41]). Moreover, M is a *minimal space*, that is to say, it admits a CW structure with as many k-cells as its k-th Betti number ([21]). The complex described in [43] realizes this structure by refining the Salvetti complex through the use of discrete Morse theory. When $m = 2$, i.e. $\mathcal{A}$ is an arrangement of affine lines in the real plane $\mathbb{R}^2$, an explicit description of the boundary operators of this complex is given in [26]. In this section we recall briefly the results of [43] and [26].

Let $\mathcal{A}$ be an arrangement of n affine hyperplanes in $\mathbb{R}^m$ and $M = \mathcal{M}(\mathcal{A}_{\mathbb{C}}) \subseteq \mathbb{C}^m$ be the complement of the complexified arrangement. If $x, y \in \mathbb{R}^m$, consider the equivalence relation $x \sim y$ if and only if for all $H \in \mathcal{A}$ either both x and y belong to H, or they are strictly on the same side of H. A *facet* is an equivalence class of this relation (see also [5]).

The *support* of a facet F is the intersection of all the hyperplanes containing F; the *(co)dimension* of a facet is the (co)dimension of its support. A *chamber* is a facet of codimension 0. In Figure 2.1, there are seven facets of codimension 0 (chambers), nine facets of codimension 1 (segments and rays), and three facets of codimension 2 (points).

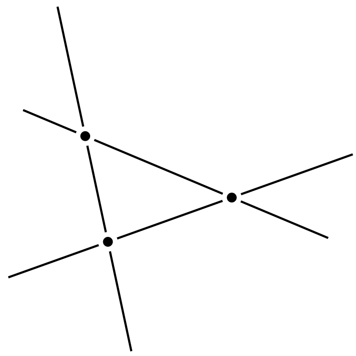


Figure 2.1: Example of stratification of $\mathbb{R}^2$ in facets.

Denote by $\mathcal{F}$ the set of all facets; it is a poset with the partial order

$$F_1 \preccurlyeq F_2 \quad \text{if and only if} \quad \overline{F_1} \supseteq F_2$$

where $\overline{F_1}$ is the topological closure. The k-cells of $\mathcal{S}(\mathcal{A})$ are in 1-1 correspondence with the set

$$\{(C, F) \in \mathcal{F} \times \mathcal{F} \mid \operatorname{codim}(C) = 0,\ \operatorname{codim}(F) = k,\ C \preccurlyeq F\}.$$

Moreover, a cell (D, G) appears in the expression of the boundary of a cell (C, F) if and only if

1. $G \preccurlyeq F$;

2. the chambers C and D are contained in the same chamber in the stratification induced by $\mathcal{A}_G$

where $\mathcal{A}_G$ is the subarrangement of $\mathcal{A}$ given by the hyperplanes containing G.

From now on, in order to simplify the description, we will consider only the case $m = 2$. Recall that a system of polar coordinates for $\mathbb{R}^2$ is a pair (O, ν) where O is a point and ν is a line containing O. If $\nu(\theta)$ is the line obtained by rotating counter-clockwise the line ν around O of an angle $\theta \in [0, 2\pi)$, then we say that a point $P \in \mathbb{R}^2 \setminus \{O\}$ has coordinates $(\rho, \theta) \in \mathbb{R}_{>0} \times [0, 2\pi)$ if it belongs to $\nu(\theta)$ and its distance from O is ρ.

Definition 2.12. Let (O, ν) be a system of polar coordinates and $\mathcal{A}$ be a line arrangement in $\mathbb{R}^2$. We say that (O, ν) is *generic* with respect to $\mathcal{A}$ if:

1. O belongs to a chamber of $\mathcal{F}$;

2. there exists $0 < \delta < \pi/2$ such that the union of the bounded facets of $\mathcal{F}$ is contained in the open positive cone

$$C(0, \delta) \coloneqq \{(\rho, \theta) \in \mathbb{R}^2 \mid 0 < \theta < \delta\};$$

3. the lines $\nu(\theta)$ with $0 \leqslant \theta \leqslant \delta$ are generic with respect to $\mathcal{A}$, i.e. for every $\ell \in \mathcal{A}$ the intersection $\ell \cap \nu(\theta)$ is a single point that belongs to $\overline{C(0, \delta)}$ (topological closure);

4. each line $\nu(\theta)$ with $0 \leqslant \theta \leqslant \delta$ contains at most one 0-dimensional facet (i.e. a point) of $\mathcal{F}$.

A generic system of polar coordinates allows us to define a total ordering on the set of facets $\mathcal{F}$.

Definition 2.13. Let $F \in \mathcal{F}$. Denote

$$\theta(F) \coloneqq \inf\{\eta \mid \nu(\eta) \cap F \neq \varnothing\} \in [0, 2\pi).$$

The *polar ordering* is the total ordering on $\mathcal{F}$ defined by: $F \lhd G$ if and only if

1. $\theta(F) < \theta(G)$;

2. $\theta(F) = \theta(G)$, F is a point and G is not a point;

3. $\theta(F) = \theta(G)$, for all $\varepsilon > 0$ the two sets $\nu(\theta(F) + \varepsilon) \cap F$ and $\nu(\theta(G) + \varepsilon) \cap G$ are not empty and there is a point belonging to the first set that is closer to the origin than all the points of the second set.

The cells of the minimal complex described in [43] are the cells of the Salvetti complex $\mathcal{S}(\mathcal{A})$ that are critical with respect to a suitable discrete Morse function, and in the 2-dimensional case there is a characterisation of them in terms of the polar ordering:

1. the critical 2-cells are the pairs (C, P) such that the point P is the maximal facet of $\mathcal{F}$ contained in $\overline{C}$ with respect to the polar ordering;
2. the critical 1-cells are the pairs (C, F) such that $F \cap \nu \neq \varnothing$ and $C \lhd F$;
3. the only critical 0-cell is (C_0, C_0) where C_0 is the chamber containing O.

Now, let $\mathcal{A}$ be an arrangement of n affine lines in $\mathbb{R}^2$ and let (R, ρ) be a local system for $\mathcal{A}$ such that for a generator β_i of $H_1(M; \mathbb{Z})$ we have $\rho(\beta_i) = t_i \in \mathcal{A}ut(R)$. Denote with $\mathcal{C}_2$, $\mathcal{C}_1$ and $\mathcal{C}_0$ be the free R-modules generated respectively by the critical 2-, 1- and 0-cells. Notice that

$$\operatorname{rk}(\mathcal{C}_0) = 1, \qquad \operatorname{rk}(\mathcal{C}_1) = n, \qquad \operatorname{rk}(\mathcal{C}_2) = \sum_{P \in \mathcal{S}ing(\mathcal{A})} (m(P) - 1)$$

where as usual $m(P)$ is the multiplicity of a singular point P. We have to describe the boundary operators $\partial_2 \colon \mathcal{C}_2 \to \mathcal{C}_1$ and $\partial_1 \colon \mathcal{C}_1 \to \mathcal{C}_0$.

Label the lines of $\mathcal{A}$ as $\ell_1, \dots, \ell_n$ depending on the order of their intersections with the reference line ν. In other words, if $d(P)$ denotes the distance of the point $P \in \mathbb{R}^2$ from the origin O, choose labels $\ell_1, \dots, \ell_n$ for the lines of $\mathcal{A}$ such that

$$d(\ell_1 \cap \nu) < \cdots < d(\ell_n \cap \nu).$$

We have to introduce a little notation; see Figure 2.2 for reference. If $P \in \mathcal{S}ing(\mathcal{A})$, denote by $S(P) \coloneqq \{\ell \in \mathcal{A} \mid P \in \ell\}$ the set of lines of $\mathcal{A}$ passing through P; let $\ell_{S(P)}$ and $\ell^{S(P)}$ be the lines in $S(P)$ with the minimum and the maximum index respectively. Moreover, let $U(P)$ be the set of the lines of $\mathcal{A}$ passing above P and $L(P)$ be the set of lines passing below P; more precisely, if $\nu(P)$ is the line passing through O and P,

$$U(P) = \{\ell \in \mathcal{A} \mid d(\nu(P) \cap \ell) > d(P)\} \text{ and } L(P) = \{\ell \in \mathcal{A} \mid d(\nu(P) \cap \ell) < d(P)\}.$$

Finally, $\operatorname{Cone}(P)$ is the closed cone delimited by $\ell_{S(P)}$ and $\ell^{S(P)}$ with vertex P and such that its intersection with the reference line ν is bounded.

For a critical 2-cell (C, P), there are exactly two lines of $S(P)$ that delimit C: denote with ℓ_C and ℓ^C the one with minimum and maximum index respectively. Define also

- $U(C) \coloneqq \{\ell_i \in S(P) \mid i \geqslant \text{index of } \ell^C\}$;
- $L(C) \coloneqq \{\ell_i \in S(P) \mid i \leqslant \text{index of } \ell_C\}$.

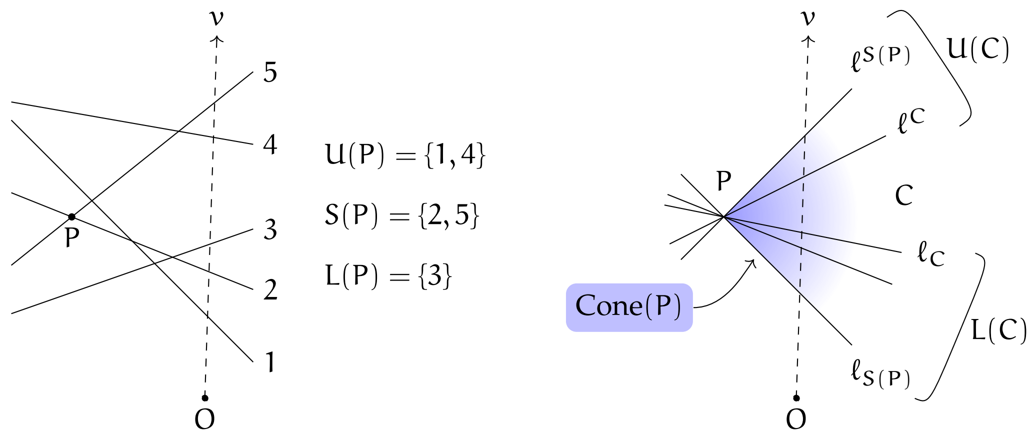


Figure 2.2: Notation for the lines with respect to the singular point P and for a critical 2-cell (C, P).

Theorem 2.14. *Let (C_0, F_1), $(C_1, F_2), \dots,$ (C_{n-1}, F_n) be the critical 1-cells, where for all $j = 1, \dots, n$ we have $F_j \subseteq \ell_j$. Then, for a critical 2-cell (C, P) and an element $h \in R$, the image of $h(C, P) \in \mathcal{C}_2$ under ∂_2 is given by*

$$\sum_{\substack{(C_{j-1},F_j) \text{ s.t.}\\ \ell_j\in S(P)}} \left(\left(\prod_{\substack{i<j \text{ s.t.}\\ \ell_i\in U(P)}} t_i\right)\left(\prod_{\substack{i \text{ s.t.}\\ \ell_i\in[C\to\ell_j)}} t_i - \prod_{\substack{i<j \text{ s.t.}\\ \ell_i\in S(P)}} t_i\right)\right)(h)(C_{j-1},F_j) +$$
$$+ \sum_{\substack{(C_{j-1},F_j) \text{ s.t.}\\ \ell_j\in U(P)\\ F_j\subseteq \mathrm{Cone}(P)}} \left(\left(\prod_{\substack{i<j \text{ s.t.}\\ \ell_i\in U(P)}} t_i\right)\left(1-\prod_{\substack{i<j \text{ s.t.}\\ \ell_i\in L(C)}} t_i\right)\left(\prod_{\substack{i<j \text{ s.t.}\\ \ell_i\in U(C)}} t_i - \prod_{\substack{i \text{ s.t.}\\ \ell_i\in U(C)}} t_i\right)\right)(h)(C_{j-1},F_j) \tag{2.2}$$

where $[C \to \ell_j)$ is the subset of $S(P)$ defined by

$$[C\to\ell_j) := \begin{cases} \{\ell_k \in U(C) \mid k<j\} & \text{if } \ell_j \in U(C); \\ \{\ell_k \in S(P) \mid k<j\} \cup U(C) & \text{if } \ell_j \in L(C). \end{cases}$$

Conventionally, an empty product is equal to 1.

Theorem 2.15. *Given a critical 1-cell (C_{i-1}, F_i) and an element $h \in R$, the image of $h(C_{i-1}, F_i) \in \mathcal{C}_1$ under ∂_1 is*

$$\partial_1(h(C_{i-1}, F_i)) = (1 - t_i)(h)(C_0, C_0).$$

Example 2.1. Consider the "deconed A3 arrangement" $\mathrm{d}A3$ in Figure 2.3. Let us try to compute the image under ∂_2 of the critical 2-cell (C, P) highlighted in blue.

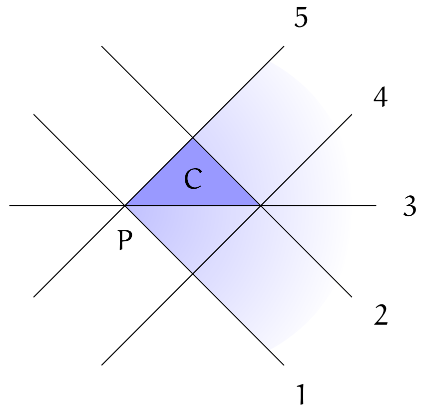


Figure 2.3: Reference figure for Example 2.1.

For the point P we have

$$S(P) = \{\ell_1, \ell_3, \ell_5\}, \quad U(P) = \{\ell_2\}, \quad L(P) = \{\ell_4\}$$

and Cone(P) is the light blue region in the figure; for the chamber C we have

$$U(C) = \{\ell_5\}, \quad L(C) = \{\ell_1, \ell_3\}.$$

From Equation (2.2) we know that the 1-cells with a non-zero coefficient in $\partial_2(C, P)$ are the ones in $S(P)$, which are lines 1, 3 and 5, and the ones in Cone(P) that pass above P—in this case all the lines belong to Cone(P), but only line 2 passes above.

- Coefficient for line 1 ($j = 1$ in Equation (2.2)): in this case $[C \to \ell_1) = \{\ell_5\}$ and
  - $\{i \mid i < 1,\ \ell_i \in U(P)\} = \varnothing$ because there are no indices below 1;
  - $\{i \mid \ell_i \in [C \to \ell_1)\} = \{5\}$;
  - $\{i \mid i < 1,\ \ell_i \in S(P)\} = \varnothing$ because there are no indices below 1;

  therefore the coefficient is $(t_5 - 1)$.

- Coefficient for line 2 ($j = 2$):
  - $\{i \mid i < 2,\ \ell_i \in U(P)\} = \varnothing$ because the only line in $U(P)$ is ℓ_2 (notice that the inequality is strict);
  - $\{i \mid i < 2,\ \ell_i \in L(C)\} = \{1\}$;
  - $\{i \mid i < 2,\ \ell_i \in U(C)\} = \varnothing$;
  - $\{i \mid \ell_i \in U(C)\} = \{5\}$;

therefore the coefficient is $(1 - t_1)(1 - t_5)$.

- Coefficient for line 3 ($j = 3$): in this case $[C \to \ell_3) = \{\ell_1, \ell_5\}$ and
  - $\{i \mid i < 3,\ \ell_i \in U(P)\} = \{2\}$;
  - $\{i \mid \ell_i \in [C \to \ell_3)\} = \{1, 5\}$;
  - $\{i \mid i < 3,\ \ell_i \in S(P)\} = \{1\}$;

  therefore the coefficient is $t_2(t_1 t_5 - t_1)$.
- Coefficient for line 5 ($j = 5$): in this case $[C \to \ell_5) = \varnothing$ and
  - $\{i \mid i < 5,\ \ell_i \in U(P)\} = \{2\}$;
  - $\{i \mid \ell_i \in [C \to \ell_5)\} = \varnothing$;
  - $\{i \mid i < 5,\ \ell_i \in S(P)\} = \{1, 3\}$;

  therefore the coefficient is $t_2(1 - t_1 t_3)$.

Putting all together we have

$$\begin{aligned}\partial_2(C, P) = (t_5 - 1)(C_0, F_1) + (1 - t_1)(1 - t_5)(C_1, F_2) +\\ + t_2(t_1 t_5 - t_1)(C_2, F_3) + t_2(1 - t_1 t_3)(C_4, F_5).\end{aligned}$$

If we consider $\mathcal{C}_2 \simeq R^6$ and $\mathcal{C}_1 \simeq R^5$, we can put all the coefficients in a 5×6 matrix with coefficients in $\mathcal{A}ut(R)$ that represents the boundary operator as a map $\partial_2 \colon R^6 \to R^5$. After a little computation, it turns out that this matrix is

$$\begin{pmatrix} 0 & t_3t_5 - 1 & t_5 - 1 & 0 & 0 & t_4 - 1 \\ t_5 - 1 & (t_1 - 1)(t_3t_5 - 1) & (t_1 - 1)(t_5 - 1) & t_3t_4 - 1 & t_4 - 1 & (t_1 - 1)(t_4 - 1) \\ 0 & t_2(1 - t_1) & t_1t_2(t_5 - 1) & 1 - t_2 & t_2(t_4 - 1) & t_2(t_1 - 1)(t_4 - 1) \\ 0 & 0 & 0 & t_3(1 - t_2) & 1 - t_2t_3 & t_2t_3(1 - t_1) \\ 1 - t_2 & t_2t_3(1 - t_1) & t_2(1 - t_1t_3) & 0 & 0 & 0 \end{pmatrix}.$$

Remark. The t_i's in the above matrix can be interpreted, for example, as indeterminates of a Laurent polynomial ring $\mathbb{C}[t_1^{\pm 1}, \dots, t_n^{\pm 1}]$ or as non-zero complex numbers. Moreover, by evaluating $t_i = T$ for all i, we can compute the homology $H_*(M; \mathbb{C}[T^{\pm 1}])$ with the action given by T-multiplication, and therefore the homology of the Milnor fibre.

Chapter 3

Characteristic Varieties

We have seen in Section 2.2 that a point $\mathbf{t} \in (\mathbb{C}^*)^n$ defines a rank-1 local system $\mathbb{C}_\mathbf{t}$ for a complex affine arrangement $\mathcal{A}$ of n lines. The next step is to study how the cohomology $H^i(\mathcal{M}(\mathcal{A}); \mathbb{C}_\mathbf{t})$ varies depending on $\mathbf{t}$. This leads to the definition of the characteristic varieties of an arrangement $\mathcal{A}$.

Characteristic varieties have been studied deeply from an algebraic-geometric point of view. Less is known about the combinatorics of them. Also, the question about whether the characteristic varieties are combinatorial in the sense of Definition 1.11 is still unanswered in full. In this chapter we recall the state of the art and we propose that an old combinatorial description of the "homogeneous components" may have, in fact, something else to tell.

3.1 The Definition

The definition of the characteristic varieties is actually very simple.

Definition 3.1. Let $\mathcal{A}$ be an affine arrangement of n hyperplanes in $\mathbb{C}^m$. For $\mathbf{t} = (t_1, \dots, t_n) \in (\mathbb{C}^*)^n$, let $\mathbb{C}_\mathbf{t}$ be the rank-1 local system associated with $\mathbf{t}$. The set

$$\mathcal{V}^i_k(\mathcal{A}) \coloneqq \{\mathbf{t} \in (\mathbb{C}^*)^n \mid \dim_\mathbb{C} H^i(\mathcal{M}(\mathcal{A}); \mathbb{C}_\mathbf{t}) \geqslant k\}$$

is called *characteristic variety* of $\mathcal{A}$ of *degree* i and *depth* k. When $i = 1$, we write just $\mathcal{V}_k(\mathcal{A}) \coloneqq \mathcal{V}^1_k(\mathcal{A})$; when also $k = 1$, we denote $\mathcal{V}(\mathcal{A}) \coloneqq \mathcal{V}_1(\mathcal{A})$.

The characteristic varieties $\mathcal{V}^i_k(\mathcal{A})$ are closed algebraic subsets of $(\mathbb{C}^*)^n$ (see for example [19, Proposition 6.1]). Actually, for $i = 1$ we can be more precise about the irreducible components of $\mathcal{V}_k(\mathcal{A})$.

Theorem 3.2 (Arapura [1])**.** *$\mathcal{V}_k(\mathcal{A})$ is a finite union of (possibly translated) subtori of the complex torus $(\mathbb{C}^*)^n$.*

Notation 3.3. The union of the components passing through $\mathbf{1} = (1, \dots, 1) \in (\mathbb{C}^*)^n$ is the "homogeneous part" of the characteristic variety $\mathcal{V}_k(\mathcal{A})$ and will be denoted by $\check{\mathcal{V}}_k(\mathcal{A})$.

3.2 Computing Characteristic Varieties

Given a finite presentation of $\pi_1(\mathcal{M}(\mathcal{A}))$, it is possible to compute the characteristic variety $\mathcal{V}(\mathcal{A})$ using Fox calculus.

Definition 3.4. Let G be a group. The *group ring* associated with G (with coefficients in $\mathbb{C}$) is the set $\mathbb{C}G$ of all finite $\mathbb{C}$-linear combinations of elements of G, equipped with the ring structure given by

$$\Big(\sum_{g\in G} m_g g\Big) + \Big(\sum_{g\in G} n_g g\Big) = \sum_{g\in G}(m_g + n_g)g,$$
$$\Big(\sum_{g\in G} m_g g\Big) \cdot \Big(\sum_{h\in G} n_h h\Big) = \sum_{g,h\in G}(m_g n_h)(gh).$$

Suppose that G is finitely generated by $g_1, \dots, g_n$. For $i = 1, \dots, n$ define the *Fox derivative* $\partial/\partial g_i$ as the $\mathbb{C}$-linear map

$$\frac{\partial}{\partial g_i} \colon \mathbb{C}G \to \mathbb{C}G$$

such that

$$\frac{\partial}{\partial g_i}(1) = 0, \qquad \frac{\partial}{\partial g_i}(g_j) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases}$$

and for all $g, h \in G$

$$\frac{\partial}{\partial g_i}(gh) = \frac{\partial}{\partial g_i}(g) + g\frac{\partial}{\partial g_i}(h).$$

Now, let $G = \langle g_1, \dots, g_n \mid r_1, \dots, r_h \rangle$ be a presentation of G. The *Fox Jacobian* of G is the matrix $J_G \in \mathcal{M}_{h\times n}(\mathbb{C}G)$ defined as

$$(J_G)_{i,j} = \frac{\partial}{\partial g_j}(r_i).$$

Suppose that $r_i \in [G, G]$ for all $i = 1, \dots, h$ (that is the case if $G = \pi_1(\mathcal{M}(\mathcal{A}))$). The abelianization ${}^{G}/_{[G,G]}$ is isomorphic to $\mathbb{Z}^n$ and the abelianization map $\mathrm{ab}\colon G \to \mathbb{Z}^n$ extends to a map

$$\mathrm{ab}\colon \mathbb{C}G \to \mathbb{C}\mathbb{Z}^n \simeq \mathbb{C}[T_1^{\pm 1}, \dots, T_n^{\pm 1}]$$

where $\mathrm{ab}(g_i) = T_i$.

Definition 3.5. The *Alexander matrix* of the group G is the abelianization of the Fox Jacobian of G, i.e. it is the matrix $A_G \in \mathcal{M}_{h\times n}\Big(\mathbb{C}[T_1^{\pm 1}, \dots, T_n^{\pm 1}]\Big)$ such that $(A_G)_{i,j} = ab((J_G)_{i,j})$.

Theorem 3.6 (see [45] for references). *Let* $G = \pi_1(\mathcal{M}(\mathcal{A}))$ *and, for* $\mathbf{t} \in (\mathbb{C}^*)^n$*, let* $A_G(\mathbf{t}) \in \mathcal{M}_{h\times n}(\mathbb{C})$ *be the Alexander matrix of* G *evaluated at* $\mathbf{t}$*. Then*

$$\mathcal{V}_k(\mathcal{A}) = \{\mathbf{t} \in (\mathbb{C}^*)^n \mid \operatorname{rk}(A_G(\mathbf{t})) < n - k\}.$$

Despite all this machinery, we will *not* use the Alexander matrix to compute the characteristic varieties. In fact, for real affine line arrangements, the transposed of the algebraic complex introduced in Section 2.3 allows us to compute directly $H^1(\mathcal{M}(\mathcal{A}_\mathbb{C}); \mathbb{C}_\mathbf{t})$. It follows easily that, if $[\partial_2](\mathbf{t})$ is the matrix of the second boundary of that complex,

$$\mathcal{V}_k(\mathcal{A}_\mathbb{C}) = \{\mathbf{t} \in (\mathbb{C}^*)^n \mid \operatorname{rk}([\partial_2](\mathbf{t})) < n - k\}.$$

This characterisation leads to a first, easy result about what happens if we add a line to an arrangement.

Proposition 3.7. *Let* $\mathcal{A} = \{\ell_1, \dots, \ell_n\}$ *be an arrangement of lines in* $\mathbb{R}^2$ *and let* $\mathcal{A}' = \mathcal{A} \cup \{\ell_{n+1}\}$*. Let* $\mathcal{V}(\mathcal{A}) \subseteq (\mathbb{C}^*)^n$ *and* $\mathcal{V}(\mathcal{A}') \subseteq (\mathbb{C}^*)^{n+1}$ *be the characteristic varieties of (the complexifications of)* $\mathcal{A}$ *and* $\mathcal{A}'$ *respectively.*

1. *If* $(t_1, \dots, t_n) \in \mathcal{V}(\mathcal{A})$*, then* $(t_1, \dots, t_n, 1) \in \mathcal{V}(\mathcal{A}')$*.*
2. *Let* $\mathcal{P} \subseteq \mathcal{S}ing(\mathcal{A}')$ *be the set of points that belong to* ℓ_{n+1} *and suppose that there exists* $P \in \mathcal{P}$*,* $P = \ell_{i_1} \cap \dots \cap \ell_{i_r} \cap \ell_{n+1}$ *such that* $t_{i_1} \cdots t_{i_r} \neq 1$*. If* $(t_1, \dots, t_n, 1) \in \mathcal{V}(\mathcal{A}')$*, then* $(t_1, \dots, t_n) \in \mathcal{V}(\mathcal{A})$*.*

Proof. Without loss of generality, we can suppose that ℓ_{n+1} is the last line of $\mathcal{A}'$ with respect to the polar ordering. We compute ∂_2 of $\mathcal{A}$ and $\mathcal{A}'$ using the Formula (2.2).

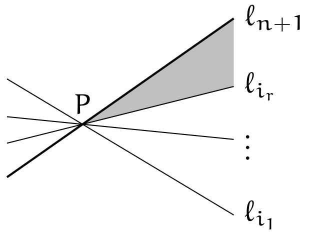


Figure 3.1: The shaded 2-cell is the last one of a singular point $P \in \mathcal{P}$, i.e. belonging to ℓ_{n+1}.

Notice that there is a bijective correspondence between the critical 2-cells of $\mathcal{A}$ and the critical 2-cells of $\mathcal{A}'$ which are *not* of the form highlighted in Figure 3.1. In particular, the coefficients given by Formula (2.2) for these cells relatively to the lines $\ell_1, \dots, \ell_n$ computed for $\mathcal{A}$ and $\mathcal{A}'$ are the same except possibly for some t_{n+1} factors, and become exactly the same if we evaluate $t_{n+1} = 1$.

On the other hand, the same formula computes the coefficients of a critical 2-cell of $\mathcal{A}'$ of the form given in Figure 3.1. It turns out that these coefficients relatively to $\ell_1, \dots, \ell_n$ either are 0 or have a factor $1 - t_{n+1}$, so they are all equal to 0 when $t_{n+1} = 1$.

In conclusion, let $D' \coloneqq [\partial_2(\mathcal{A}')](t_1, \dots, t_n, 1)$ and $D \coloneqq [\partial_2(\mathcal{A})](t_1, \dots, t_n)$. Up to moving the columns relative to the 2-cells of the form of Figure 3.1 at the end, we have

$$D' = \left(\begin{array}{c|ccc} & | & & | \\ D & 0 & \cdots & 0 \\ & | & & | \\ \hline (**) & & (*) & \end{array} \right).$$

The coefficients in the part $(*)$ of the matrix can also be computed: if $P \in \mathcal{P}$ is the singular point corresponding to the 2-cell, the relative coefficient is

$$\left(\prod_{\ell_i \in U(P)} t_i \right) \cdot \left(1 - \prod_{i \in P} t_i \right).$$

1. Suppose that $(t_1, \dots, t_n) \in \mathcal{V}(\mathcal{A})$, i.e. all $(n-1) \times (n-1)$ minors of D vanish. We want to prove that all $n \times n$ minors of D' also vanish. To do so, we analyse all the possible choices of such a minor.

   - If we choose the first n rows of D',
     - if we choose n columns of D then the minor is zero because $\mathrm{rk}(D) \leqslant n-1$;
     - if we choose at least one column of zeros, the minor is trivially zero.
   - if we choose the last row and $n-1$ among the others, we use Laplace expansion on the last row to obtain a sum of $(n-1) \times (n-1)$ minors (with some coefficients) that involve the first n rows; we now prove that all these smaller minors are 0.
     - If the $n-1$ columns are chosen among the ones of D, the minor is zero because $(t_1, \dots, t_n) \in \mathcal{V}(\mathcal{A})$;
     - if we choose at least one column of zeros, the minor is trivially zero.

   Therefore all the $n \times n$ minors of D' are zero, and $(t_1, \dots, t_n, 1) \in \mathcal{V}(\mathcal{A}')$.

2. Suppose now that $(t_1, \dots, t_n, 1) \in \mathcal{V}(\mathcal{A}')$, i.e. all the $n \times n$ minors of D' vanish. Choose an $(n-1) \times (n-1)$ minor M of D, and complete it with the last row of

D and the column corresponding to a $P \in \mathcal{P}$, $P = \ell_{i_1} \cap \cdots \cap \ell_{i_r} \cap \ell_{n+1}$, such that $t_{i_1} \cdots t_{i_r} \neq 1$ (it exists by hypothesis). The resulting $n \times n$ minor of D' is

$$M \cdot \left(\prod_{\ell_i \in U(P)} t_i \right) \cdot \left(1 - \prod_{j=1}^{r} t_{i_j} \right)$$

and it is equal to 0. Now,

- $\prod_{\ell_i \in U(P)} t_i \neq 0$ because they are invertible elements;
- $t_{i_1} \cdots t_{i_r} \neq 1$;

we can conclude that $M = 0$. □

Remark. The implication $(t_1, \dots, t_n, 1) \in \mathcal{V}(\mathcal{A}') \Rightarrow (t_1, \dots, t_n) \in \mathcal{V}(\mathcal{A})$ is false in general, without adding further hypotheses. For example, consider the two arrangements in Figure 3.2.

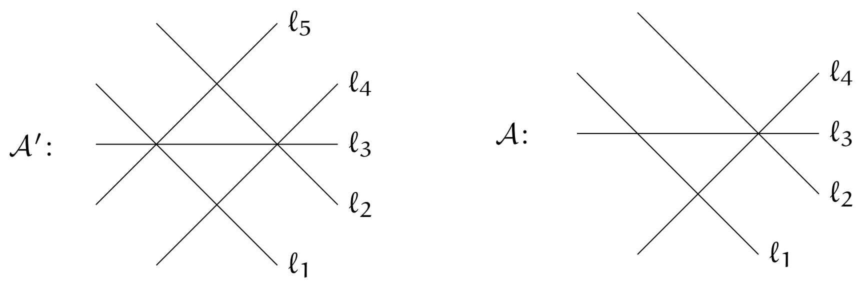


Figure 3.2

- $\mathcal{V}(\mathcal{A}') = V_1' \cup \cdots \cup V_5'$ where:

$$\begin{aligned} V_1' &= \{\mathbf{t} \in (\mathbb{C}^*)^5 \mid t_3 = 1,\ t_4 = 1,\ t_5 = 1\}, \\ V_2' &= \{\mathbf{t} \in (\mathbb{C}^*)^5 \mid t_4 = 1,\ t_2 = 1,\ t_1 t_3 t_5 = 1\}, \\ V_3' &= \{\mathbf{t} \in (\mathbb{C}^*)^5 \mid t_2 = t_5,\ t_1 = t_4,\ t_3 t_4 t_5 = 1\}, \\ V_4' &= \{\mathbf{t} \in (\mathbb{C}^*)^5 \mid t_5 = 1,\ t_1 = 1,\ t_2 t_3 t_4 = 1\}, \\ V_5' &= \{\mathbf{t} \in (\mathbb{C}^*)^5 \mid t_1 = 1,\ t_2 = 1,\ t_3 = 1\}; \end{aligned}$$

- $\mathcal{V}(\mathcal{A}) = V_1 \cup V_2$ where:

$$\begin{aligned} V_1 &= \{\mathbf{t} \in (\mathbb{C}^*)^4 \mid t_3 = 1,\ t_4 = 1\}, \\ V_2 &= \{\mathbf{t} \in (\mathbb{C}^*)^4 \mid t_1 = 1,\ t_2 t_3 t_4 = 1\}. \end{aligned}$$

Now, the point $(1,1,1,a,1)$ belongs to V_5' for every $a \in \mathbb{C}^*$, but if $a \neq 1$ the point $(1,1,1,a)$ does not belong to $\mathcal{V}(\mathcal{A})$. Notice that $\mathcal{P} = \{\ell_1 \cap \ell_3 \cap \ell_5, \ell_2 \cap \ell_5\}$ and for the point $(1,1,1,a,1)$ we have both $t_2 = 1$ and $t_1 t_3 = 1$.

3.3 Resonance Varieties

It is known that the homogeneous part $\check{\mathcal{V}}_k(\mathcal{A})$ of the characteristic variety *is* combinatorial: it can be derived by the so-called resonance variety. Let $\mathcal{A}$ be an arrangement of n affine hyperplanes in $\mathbb{C}^m$ and let $A = A(\mathcal{A})$ be the Orlik-Solomon algebra of $\mathcal{A}$. Recall that A is a graded algebra $A = \bigoplus A_i$ and it is combinatorial. Let $a \in A_1$ and consider the map "multiplication by a"

$$\begin{array}{rccc} \delta_a\colon & A & \longrightarrow & A \\ & b & \longmapsto & a \wedge b. \end{array}$$

Since $\delta_a(A_i) \subseteq A_{i+1}$ and $a \wedge a = 0$, $(A_\bullet, \delta_a)$ is a cochain complex, called *Aomoto complex*.

Definition 3.8. The set

$$\mathcal{R}^i_k(\mathcal{A}) \coloneqq \{a \in A_1 \mid \dim_{\mathbb{C}} H^i((A_\bullet, \delta_a); \mathbb{C}) \geqslant k\}$$

is called *resonance variety* of $\mathcal{A}$ of *degree* i and *depth* k. When $i = 1$, we write just $\mathcal{R}_k(\mathcal{A}) \coloneqq \mathcal{R}^1_k(\mathcal{A})$; when also $k = 1$, we denote $\mathcal{R}(\mathcal{A}) \coloneqq \mathcal{R}_1(\mathcal{A})$.

The relation between characteristic varieties and resonance varieties is established by the following theorem.

Theorem 3.9 (Tangent Cone [8])**.** *The resonance variety $\mathcal{R}_k(\mathcal{A})$ is the tangent cone of the characteristic variety $\mathcal{V}_k(\mathcal{A})$ at the point $\mathbf{1} = (1,\dots,1)$. In particular, the exponential map*

$$\begin{array}{rccc} \exp\colon & A_1[\simeq \mathbb{C}^n] & \longrightarrow & (\mathbb{C}^*)^n \\ & (z_1,\dots,z_n) & \longmapsto & (e^{2\pi i z_1},\dots,e^{2\pi i z_n}) \end{array}$$

defines a 1-1 correspondence between the set of irreducible components of $\mathcal{R}_k(\mathcal{A})$ and the set of irreducible components of $\mathcal{V}_k(\mathcal{A})$ passing through $\mathbf{1}$.

Resonance varieties have been extensively studied by several authors. Here we recall some of their properties.

Proposition 3.10 ([23])**.** *If $\mathcal{A}$ is central, then $\mathcal{R}_1(\mathcal{A})$ is contained in the subspace*

$$\left\{ (z_1,\dots,z_n) \in \mathbb{C}^n \;\middle|\; \sum_{i=1}^n z_i = 0 \right\}.$$

Proposition 3.11 ([8]). *The irreducible components of* $\mathcal{R}_1(\mathcal{A})$ *are linear subspaces of* $\mathbb{C}^n$.

Proposition 3.12 ([34]). *Let* $\mathcal{C}$ *be the set of irreducible components of* $\mathcal{R}_1(\mathcal{A})$.

1. *For all* $C \in \mathcal{C}$, $\dim C \geqslant 2$.
2. *For all* $C_1, C_2 \in \mathcal{C}$, *if* $C_1 \neq C_2$ *then* $C_1 \cap C_2 = \{0\}$.
3. *Let* $\mathcal{C}_k := \{C \in \mathcal{C} \mid \dim C \geqslant k+1\}$. *Then for all* k

$$\mathcal{R}_k(\mathcal{A}) = \{0\} \cup \bigcup_{C \in \mathcal{C}_k} C.$$

Proposition 3.13 ([19, Corollary 6.2]). *The irreducible components of* $\mathcal{R}_1(\mathcal{A})$ *are precisely the maximal isotropic subspaces* $C \subseteq A_1 \left[\, \simeq H^1(\mathcal{M}(\mathcal{A}); \mathbb{C})\right]$ *with respect to the cup product*

$$\cup\colon\ H^1(\mathcal{M}(\mathcal{A}); \mathbb{C}) \times H^1(\mathcal{M}(\mathcal{A}); \mathbb{C}) \to H^2(\mathcal{M}(\mathcal{A}); \mathbb{C})$$

and such that $\dim C \geqslant 2$.

3.4 The Setting

From now on, $\mathcal{A}$ will be a *projective* line arrangement of $n+1$ lines in $\mathbb{P}^2(\mathbb{R})$. Notice that we have not defined local systems for projective arrangements. However, it is known that, if $\beta_1, \dots, \beta_{n+1}$ are the generators of $H_1(\mathcal{M}(\mathcal{A}_{\mathbb{C}}); \mathbb{Z})$ (i.e. geometric loops around the projective lines), then

$$H_1(\mathcal{M}(\mathcal{A}_{\mathbb{C}}); \mathbb{Z}) = \langle \beta_1, \dots, \beta_{n+1} \mid \beta_1 \cdots \beta_{n+1} = 1,\ [\beta_i, \beta_j] = 1 \text{ for all } i, j \rangle$$

where $[\beta_i, \beta_j]$ is the commutator. It follows that, if $\mathbb{C}_{\mathbf{t}}$ is a local system for a$\mathcal{A}$ (where ℓ_{n+1} is the line sent to infinity) defined by $\mathbf{t} = (t_1, \dots, t_n)$, then $(t_1, \dots, t_{n+1})$ defines a local system for $\mathcal{A}$ if we set $t_{n+1} = (t_1 \cdots t_n)^{-1}$.

On the other hand, for a central arrangement of $n+1$ hyperplanes in $\mathbb{C}^m$ we have (see for example [8])

$$\mathcal{M}(\mathcal{A}) = \mathcal{M}(\mathrm{d}\mathcal{A}) \times \mathbb{C}^*,$$

thus $\pi_1(\mathcal{M}(\mathcal{A})) = \pi_1(\mathcal{M}(\mathrm{d}\mathcal{A})) \times \mathbb{Z}$ where the generator of the factor $\mathbb{Z}$ can be taken as the product $\beta_1 \cdots \beta_{n+1}$ where the β_i are geometric loops around the hyperplanes of $\mathcal{A}$. It follows that

$$\mathcal{V}_k(\mathcal{A}) = \{(t_1, \dots, t_{n+1}) \in (\mathbb{C}^*)^{n+1} \mid (t_1, \dots, t_n) \in \mathcal{V}_k(\mathrm{d}\mathcal{A}),\ t_1 \cdots t_{n+1} = 1\}. \tag{3.1}$$

As a consequence, we *define* the characteristic variety $\mathcal{V}(\mathcal{A})$ for a projective line arrangement in $\mathbb{P}^2(\mathbb{R})$ as the characteristic variety of the complexified deprojectivization of $\mathcal{A}$:

$$\mathcal{V}(\mathcal{A}) := \mathcal{V}(\widehat{\mathcal{A}}_{\mathbb{C}}).$$

In particular we can restate the result (3.1).

Theorem 3.14. *Let $\mathcal{A} = \{\ell_1, \dots, \ell_{n+1}\}$ be a projective line arrangement of $n+1$ lines in $\mathbb{P}^2(\mathbb{R})$ and let $\mathrm{a}\mathcal{A}$ be any arrangement of n lines in $\mathbb{R}^2$ obtained by sending to infinity one of the lines of $\mathcal{A}$ (suppose, without loss of generality, ℓ_{n+1}). Then*

$$\mathcal{V}(\mathcal{A}) = \{(t_1, \dots, t_{n+1}) \in (\mathbb{C}^*)^{n+1} \mid (t_1, \dots, t_n) \in \mathcal{V}((\mathrm{a}\mathcal{A})_{\mathbb{C}}),\ t_1 \cdots t_{n+1} = 1\}.$$

Remark. From now on, we will drop the notation $\mathcal{V}(\mathcal{A}_{\mathbb{C}})$ and we will use simply $\mathcal{V}(\mathcal{A})$ to denote the characteristic variety of $\mathcal{A}$. If $\mathcal{A}$ is a real arrangement, $\mathcal{V}(\mathcal{A})$ will always denote the characteristic variety of the complexified arrangement. Similarly $\mathcal{R}(\mathcal{A})$ will be the resonance variety of $\mathcal{A}_{\mathbb{C}}$ if $\mathcal{A}$ is real.

Notation 3.15. If $\mathcal{A} = \{\ell_1, \dots, \ell_{n+1}\}$ is a projective line arrangement, we define $L_2(\mathcal{A})$ as the set of its singular points, where a point $P \in \mathcal{S}ing(\mathcal{A})$ is identified with the set of indices of the lines passing through it. Formally,

$$L_2(\mathcal{A}) \coloneqq \{\{i \mid P \in \ell_i\} \mid P \in \mathcal{S}ing(\mathcal{A})\}.$$

Abusing notation, we will use the same symbol P to denote a singular point in $\mathcal{S}ing(\mathcal{A})$ and the corresponding set of indices in $L_2(\mathcal{A})$. In a similar way we will sometimes write $\#(P)$ instead of $m(P)$.

3.5 Combinatorics of the Resonance Variety

We know that the resonance variety $\mathcal{R}(\mathcal{A})$ is combinatorially determined, because it is defined in terms of the Orlik-Solomon algebra, but we would like to have a simpler combinatorial description of its irreducible components. Falk [23] offers such a description.

Proposition 3.16 ([23, Lemma 3.14])**.** *Let $P \in L_2(\mathcal{A})$ with $\#(P) \geqslant 3$. Then*

$$C(P) \coloneqq \left\{ z \in \mathbb{C}^{n+1} \;\middle|\; \sum_{j=1}^{n+1} z_j = 0 \right\} \cap \bigcap_{j \notin P} \{z \in \mathbb{C}^{n+1} \mid z_j = 0\} \tag{3.2}$$

is an irreducible component of $\mathcal{R}(\mathcal{A})$ of dimension $\#(P) - 1$.

Definition 3.17. A component of $\mathcal{R}(\mathcal{A})$ of the form $C(P)$ for some $P \in L_2(\mathcal{A})$ with $\#(P) \geqslant 3$ is called *local component*.

The non-local components admit a description in terms of the so-called neighbourly partitions.

Definition 3.18 ([23])**.** Let $\mathcal{A}$ be an arrangement of $n+1$ lines in $\mathbb{P}^2(\mathbb{R})$. A partition $\pi = \{\pi_1, \dots, \pi_k\}$ of $[n+1]$ is *neighbourly* if, for all $P \in L_2(\mathcal{A})$ and for all $i = 1, \dots, k$,

$$\#(\pi_i \cap P) \geqslant \#(P) - 1 \;\Rightarrow\; P \subseteq \pi_i. \tag{3.3}$$

The partition with only one block $\{\{1,\ldots,n+1\}\}$ is always neighbourly, and it is the *trivial neighbourly partition*. In the following, all neighbourly partitions are implicitly assumed to be non-trivial.

Remark. Notice that if π is a neighbourly partition, the two lines passing through any double point of $\mathcal{A}$ must belong to the same block of π. It follows that any neighbourly partition is a superpartition of the double points partition.

Definition 3.19. Let $\mathcal{A}$ be an (affine or projective) arrangement. The *double points partition* of $\mathcal{A}$ is the partition induced by the connected components of the double points graph $\Gamma(\mathcal{A})$ (see Definition 2.8).

Remark. The double points partition is not necessarily neighbourly.

If π is a neighbourly partition, define $C(\pi) \subseteq \mathbb{C}^{n+1}$ as

$$C(\pi) := \left\{ z \in \mathbb{C}^{n+1} \;\middle|\; \sum_{j=1}^{n+1} z_j = 0 \right\} \cap \bigcap_{P \in \mathcal{P}} \left\{ z \in \mathbb{C}^{n+1} \;\middle|\; \sum_{j \in P} z_j = 0 \right\} \tag{3.4}$$

where $\mathcal{P} := \{P \in L_2(\mathcal{A}) \mid \nexists\, \pi_h \in \pi \text{ s.t. } P \subseteq \pi_h\}$.

Proposition 3.20 ([34])**.** *If* $\dim(C(\pi)) \geqslant 2$, *then* $C(\pi)$ *is a non-local component of* $\mathcal{R}(\mathcal{A})$.

Definition 3.21. If π is a partition of a subset $B \subseteq [n+1]$, define *support* of π, $\mathrm{supp}(\pi)$, the set B. A partition is called *essential* if $\mathrm{supp}(\pi) = [n+1]$.

Proposition 3.22 (See also [34])**.** *Let $\mathcal{B} \subseteq \mathcal{A}$ be a subarrangement and let π be a neighbourly partition for $\mathcal{B}$ such that* $\dim(C(\pi)) \geqslant 2$. *Then*

$$C(\pi) \cap \bigcap_{j \notin \mathrm{supp}(\pi)} \{z_j = 0\}$$

is a non-local component of $\mathcal{R}(\mathcal{A})$. All non-local components of $\mathcal{R}(\mathcal{A})$ arise from subarrangements of $\mathcal{A}$ this way.

Remark. In fact, even local components arise from neighbourly partitions: for $P \in L_2(\mathcal{A})$, if $\mathcal{B}_P := \{\ell \in \mathcal{A} \mid P \in \ell\}$ is the central arrangement defined by the lines passing through P, then $\pi_P := \{\{i\} \mid i \in P\}$ is neighbourly for $\mathcal{B}_P$, and $C(P) = C(\pi_P)$.

Definition 3.23. A neighbourly partition of a subarrangement $\mathcal{B} \subseteq \mathcal{A}$ is *non-local* is $\mathcal{B}$ is not of the form $\mathcal{B}_P$ for some $P \in \mathcal{S}ing(\mathcal{A})$.

A more refined combinatorial structure that describes the components of $\mathcal{R}(\mathcal{A})$ is the multinet, defined by Falk and Yuzvinsky in [24].

Definition 3.24. A *multi-arrangement* is a pair $(\mathcal{A}, \mu)$ where $\mathcal{A}$ is a line arrangement and $\mu\colon \mathcal{A} \to \mathbb{N} \setminus \{0\}$ is a multiplicity function.

Definition 3.25 ([24]). A *weak* (k,d)-*multinet* on a multi-arrangement $(\mathcal{A},\mu)$ is a pair $(\mathcal{N},\mathcal{X})$, where $\mathcal{N}=\{\mathcal{A}_1,\dots,\mathcal{A}_k\}$ is a partition of $\mathcal{A}$ in $k\geqslant 3$ classes and $\mathcal{X}\subseteq \mathcal{S}ing(\mathcal{A})$ such that

1. the value $d=\sum_{\ell\in\mathcal{A}_i}\mu(\ell)$ does not depend on i;
2. for every $\ell\in\mathcal{A}_i$ and $\ell'\in\mathcal{A}_j$ with $i\neq j$, the point $\ell\cap\ell'$ is in $\mathcal{X}$;
3. for each $P\in\mathcal{X}$, the value $n_P=\sum_{\ell\in\mathcal{A}_i,\ P\in\ell}\mu(\ell)$ does not depend on i.

A (k,d)-*multinet* is a weak (k,d)-multinet satisfying the additional property:

4. for every $\mathcal{A}_i$, for every $\ell,\ell'\in\mathcal{A}_i$ there is a sequence of lines $\ell=\ell_0,\ell_1,\dots,\ell_r=\ell'$ such that $\ell_{j-1}\cap\ell_j\notin\mathcal{X}$ for all $j=1,\dots,r$.

Remark.

- If $(\mathcal{N},\mathcal{X})$ is a weak multinet, then $\mathcal{X}$ is determined by $\mathcal{N}$, namely

$$\mathcal{X}=\{\ell\cap\ell'\mid \ell\in\mathcal{A}_i,\ \ell'\in\mathcal{A}_j,\ i\neq j\}.$$

- If $(\mathcal{N},\mathcal{X})$ is a multinet, then $\mathcal{N}$ is determined by $\mathcal{X}$ as well: build a graph G with vertex set $\mathcal{A}$ and edges $\{\ell,\ell'\}$ if and only if $\ell\cap\ell'\notin\mathcal{X}$. Then $\mathcal{N}$ is the partition induced by the connected components of G.
- If $n_P=1$ for all $P\in\mathcal{X}$, then condition 4. follows from 3. In this case $(\mathcal{N},\mathcal{X})$ is called a *net*. If $(\mathcal{N},\mathcal{X})$ is a net, $\mu(\ell)=1$ for all $\ell\in\mathcal{A}$; the converse is false, that is to say, there are multinets with $\mu(\ell)=1$ for all $\ell\in\mathcal{A}$ that are not nets.
- If $(\mathcal{N},\mathcal{X})$ is a weak multinet, $\mathcal{N}$ is neighbourly.

Theorem 3.26 ([24]). *Suppose that $\mathcal{A}$ supports a weak (k,d)-multinet. Then there is a $(k-1)$-dimensional irreducible component in $\mathcal{R}(\mathcal{A})$.*

Theorem 3.27 ([24]). *Suppose that $\mathcal{R}(\mathcal{A})$ contains a $(k-1)$-dimensional irreducible component which is not contained in any coordinate hyperplane (i.e. hyperplanes of the form $\{z_j=0\}$). Then $\mathcal{A}$ supports a (k,d)-multinet for some d.*

3.6 Combinatorics of the Characteristic Variety

The Tangent Cone Theorem establishes a correspondence between the components of the resonance variety and the ones of the characteristic variety passing through $\mathbf{1}$. Therefore we can translate the results of Section 3.5 in the context of characteristic varieties.

We begin with some classification of the irreducible components of $\mathcal{V}(\mathcal{A})$, induced by Definition 3.17.

Definition 3.28. A *local component* of $\mathcal{V}(\mathcal{A})$ is a component that corresponds to a local component of $\mathcal{R}(\mathcal{A})$.

In particular, for each $P \in L_2(\mathcal{A})$ with $\#(P) \geqslant 3$ we have a local component in the characteristic variety. Among the non-local components we make a further distinction.

Definition 3.29. A component of $\mathcal{V}(\mathcal{A})$ is *global* if it is not contained in any coordinate hypertorus of $(\mathbb{C}^*)^{n+1}$, i.e. any hypertorus defined by $\{t_i = 1\}$. Notice that a global component is always non-local (unless $\mathcal{A}$ is trivial), but there may exist components that are neither local nor global.

For $i = 1, \dots, n+1$ let $\mathcal{A}_i \coloneqq \mathcal{A} \smallsetminus \{\ell_i\}$ and let M, M_i be the complements of $\mathcal{A}_\mathbb{C}$ and $(\mathcal{A}_i)_\mathbb{C}$ respectively. The inclusion $\iota_i \colon M \hookrightarrow M_i$ induces a map of local systems

$$\begin{array}{cccc} (\iota_i)^* \colon & (\mathbb{C}^*)^n & \longrightarrow & (\mathbb{C}^*)^{n+1} \\ & (t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_{n+1}) & \longmapsto & (t_1, \dots, t_{i-1}, 1, t_{i+1}, \dots, t_{n+1}). \end{array} \tag{3.5}$$

Proposition 3.30 (Corollary of Proposition 3.7(1))**.** *If* $\mathbf{t} \in \mathcal{V}(\mathcal{A}_i)$*, then* $(\iota_i)^*(\mathbf{t}) \in \mathcal{V}(\mathcal{A})$*. In particular* $(\iota_i)^*$ *restricts to an injective map*

$$(\iota_i)^*\big|_{\mathcal{V}(\mathcal{A}_i)} \colon \mathcal{V}(\mathcal{A}_i) \to \mathcal{V}(\mathcal{A}).$$

Corollary 3.31. *Let* $\mathcal{B} \subseteq \mathcal{A}$ *be a subarrangement and let* $(\iota_\mathcal{B})^* \colon \mathcal{V}(\mathcal{B}) \to \mathcal{V}(\mathcal{A})$ *be the obvious generalization of* (3.5)*. If* W *is a component of* $\check{\mathcal{V}}(\mathcal{B})$*, then* $(\iota_\mathcal{B})^*(W)$ *is a component of* $\check{\mathcal{V}}(\mathcal{A})$*.*

Proof. By Proposition 3.30, $(\iota_\mathcal{B})^*(W) \subseteq \mathcal{V}(\mathcal{A})$—more precisely $(\iota_\mathcal{B})^*(W) \subseteq \check{\mathcal{V}}(\mathcal{A})$. The only problem that may arise is that $(\iota_\mathcal{B})^*(W) \subseteq W'$ for another component $W' \in \check{\mathcal{V}}(\mathcal{A})$. This is impossible by Proposition 3.12(2). □

Definition 3.32. A component W of $\mathcal{V}(\mathcal{A})$ is *essential* if it is not of the form $(\iota_i)^*(W_i)$ for some component W_i of $\mathcal{V}(\mathcal{A}_i)$.

Proposition 3.33. *A global component is essential.*

Proof. Let W be a global component, and suppose that $W = (\iota_i)^*(W_i)$ for some i. This would imply $W \subseteq \{t_i = 1\}$, against the hypothesis. □

Theorem 3.34 ([33, Lemma 1.4.3])**.** *A (strictly) positive-dimensional essential component is global.*

The previous result applies to all the homogeneous components of $\check{\mathcal{V}}(\mathcal{A})$, since their dimension is at least two (see Proposition 3.12(1)). As mentioned in the statement of Theorem 3.2, it is possible that an irreducible component of $\mathcal{V}(\mathcal{A})$ does not pass through the origin.

Definition 3.35. A component W of $\mathcal{V}(\mathcal{A})$ is *translated* if $(1, \dots, 1) \notin W$.

Notice that all the components of the resonance variety are non-translated (Proposition 3.11), and translated components appear in characteristic varieties (see Chapter 5). Strangely enough, all known translated components are either 0- or 1-dimensional. Theorem 3.34 does not rule out the possibility of existence of a zero-dimensional translated essential component which is non-global, however no examples of this kind have been found.

Remark. For translated components, Corollary 3.31 does not hold. In fact, let W be a translated component of $\mathcal{V}(\mathcal{B})$ for a subarrangement $\mathcal{B} \subseteq \mathcal{A}$. Then, either $(\iota_{\mathcal{B}})^*(W)$ is a (translated) component of $\mathcal{V}(\mathcal{A})$ or $(\iota_{\mathcal{B}})^*(W) \subseteq W'$ for some component $W' \subseteq \mathcal{V}(\mathcal{A})$. For example (see Chapter 5 for reference), the B3 arrangement contains 3 subarrangements of type B3x, but none of their translated components is "expressed" in the characteristic variety of B3: all these components are actually included in the 2-dimensional essential (non-translated) component of B3.

For *homogeneous* components, we can derive equations from singular points and neighbourly partitions by exponentiating Formulae (3.2) and (3.4) respectively. In particular, we define *ideals* $I \subseteq \mathbb{C}[T_1^{\pm 1}, \dots, T_{n+1}^{\pm 1}]$ such that their varieties $\mathcal{Z}(I) \subseteq (\mathbb{C}^*)^{n+1}$ are the components of $\mathcal{V}(\mathcal{A})$.

- If $P \in L_2(\mathcal{A})$ with $\#(P) \geqslant 3$, define
$$\mathcal{I}(P) \coloneqq \left(\prod_{j=1}^{n+1} T_j - 1 \right) + \big(T_j - 1 \mid j \notin P \big); \tag{3.6}$$
this corresponds to a local component of $\mathcal{V}(\mathcal{A})$.

- If $\pi = \{\pi_1, \dots, \pi_k\}$ is a neighbourly partition, define
$$\mathcal{I}(\pi) \coloneqq \left(\prod_{j=1}^{n+1} T_j - 1 \right) + \left(\prod_{j \in P} T_j - 1 \;\middle|\; P \in \mathcal{P} \right) \tag{3.7}$$
where $\mathcal{P} \coloneqq \{P \in L_2(\mathcal{A}) \mid \nexists\, \pi_h \in \pi \text{ s.t. } P \subseteq \pi_h\}$.

We can now restate Propositions 3.20 and 3.22 in the context of characteristic varieties.

Proposition 3.36. *Let W be the irreducible component of $\mathcal{Z}(\mathcal{I}(\pi))$ passing through $(1, \dots, 1)$. If $\dim(W) \geqslant 2$, then W is a non-local component of $\mathcal{V}(\mathcal{A})$.*

Proposition 3.37. *Let $\mathcal{B} \subseteq \mathcal{A}$ be a subarrangement and let π be a neighbourly partition for $\mathcal{B}$ such that $\dim(W) \geqslant 2$, where W is the component of $\mathcal{I}(\pi)$ passing through $(1, \dots, 1)$. Then the component passing through $(1, \dots, 1)$ of the variety in $(\mathbb{C}^*)^{n+1}$ defined by the ideal*
$$\mathcal{I}(\pi) + (T_j - 1 \mid j \notin \operatorname{supp}(\pi))$$
is a non-local component of $\mathcal{V}(\mathcal{A})$. All non-local components of $\mathcal{V}(\mathcal{A})$ passing through $(1, \dots, 1)$ arise from subarrangements of $\mathcal{A}$ this way.

3.7 Experiments

We used the algorithms of Sections 4.2 and 4.3 to compute characteristic varieties of several arrangements. Most of them are taken from the catalogue of simplicial arrangements of Grünbaum ([28]).

Definition 3.38. An arrangement of lines in $\mathbb{P}^2(\mathbb{R})$ is *simplicial* if the connected components of $\mathcal{M}(\mathcal{A}) \subseteq \mathbb{P}^2(\mathbb{R})$ are simplices (i.e. triangles).

We used the result to compile our small catalogue of remarkable arrangements (Chapter 5). In this section we follow the nomenclature of arrangements from that chapter.

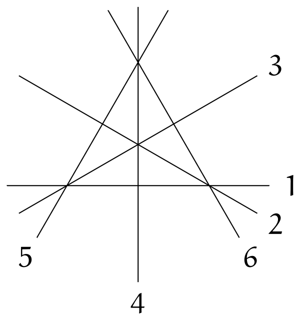


Figure 3.3: The A3 arrangement.

Let us begin with a warm-up example: the A3 arrangement. Refer to Figure 3.3 for the names of the lines. Just for this arrangement, we will explain the combinatorics of the characteristic variety with all details.

A3 has seven singular points, three double ones:

$$\{1,4\}, \quad \{2,5\}, \quad \{3,6\}$$

and four triple ones:

$$\{1,2,6\}, \quad \{1,3,5\}, \quad \{2,3,4\}, \quad \{4,5,6\}.$$

These four points give four local components

$$\begin{aligned}
&\{\mathbf{t} \in (\mathbb{C}^*)^6 \mid t_1t_2t_6 = 1,\ t_3 = 1,\ t_4 = 1,\ t_5 = 1\},\\
&\{\mathbf{t} \in (\mathbb{C}^*)^6 \mid t_1t_3t_5 = 1,\ t_2 = 1,\ t_4 = 1,\ t_6 = 1\},\\
&\{\mathbf{t} \in (\mathbb{C}^*)^6 \mid t_2t_3t_4 = 1,\ t_1 = 1,\ t_5 = 1,\ t_6 = 1\},\\
&\{\mathbf{t} \in (\mathbb{C}^*)^6 \mid t_4t_5t_6 = 1,\ t_1 = 1,\ t_2 = 1,\ t_3 = 1\}.
\end{aligned}$$

The only non-local neighbourly partition is essential, and it is the double points partition $\pi = \{\{1,4\},\{2,5\},\{3,6\}\}$. In this case

$$\mathcal{I}(\pi) = (T_1T_2T_6 - 1,\ T_1T_3T_5 - 1,\ T_2T_3T_4 - 1,\ T_4T_5T_6 - 1,\ T_1T_2T_3T_4T_5T_6 - 1)$$

is 2-dimensional and gives a global component

$$\{\mathbf{t} \in (\mathbb{C}^*)^6 \mid t_4t_5t_6 = 1,\ t_1 = t_4,\ t_2 = t_5,\ t_3 = t_6\}.$$

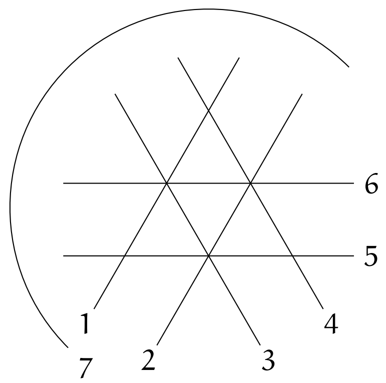


Figure 3.4: The NonFano arrangement. Line 7 is at infinity.

The next arrangement that we show is the NonFano arrangement. It has six singular points, all triple:

$$\{1,2,7\},\quad \{1,3,6\},\quad \{2,3,5\},\quad \{2,4,6\},\quad \{3,4,7\},\quad \{5,6,7\},$$

and three subarrangements of type A3. The characteristic variety has nine components, corresponding to these data. However, there is one more essential neighbourly partition:

$$\pi = \{\{1,4,5\},\{2\},\{3\},\{6\},\{7\}\}$$

(which is, indeed, the double points partition). Now, $\mathcal{I}(\pi)$ is 0-dimensional, so it does not contribute to the characteristic variety. But look at the *second* characteristic variety $\mathcal{V}_2(\text{NonFano})$ (first computed in [8]):

$$\mathcal{V}_2(\text{NonFano}) = \{(1,1,1,1,1,1,1),(1,-1,-1,1,1,-1,-1)\}.$$

There is a translated 0-dimensional component in $\mathcal{V}_2(\text{NonFano})$, and the distribution of the 1's and -1's is suspicious—it mirrors the blocks of the partition π.

The last example that we recall from literature is that of B3x. Its characteristic variety has been computed in [46] and presents one 1-dimensional translated component. This arrangement does *not* have any essential (non-trivial) neighbourly partition: to show this, begin with the double points partition

$$\{\{1,4,5\},\{2,3,7\},\{6\},\{8\}\}$$

and notice that it is not neighbourly, because for example the singular point $\{2,3,5\}$ does not respect the neighbourly condition (3.3). Since 2 and 3 are in the same block, 5

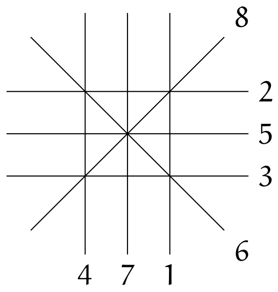


Figure 3.5: The B3x arrangement.

has to be as well, therefore $\{1, 2, 3, 4, 5, 7\}$ must be a block in the partition. Now notice that lines 6 and 8 must belong to this block too (look at the singular points $\{2, 4, 6\}$ and $\{1, 2, 8\}$) and conclude that the only essential neighbourly partition is the trivial one. It seems that in this case neighbourly partitions do not encode information of the translated component.

One of the problems of studying the translated components of characteristic varieties is the lack of examples. These components seem to appear in arrangements whose characteristic varieties are on the edge of computability. Cohen [7] has defined a class of *complex* arrangements $\mathcal{D}_r$ ($r \geqslant 2$) with $3r+2$ lines, such that $\mathcal{V}(\mathcal{D}_r)$ has $r-1$ essential 1-dimensional translated components. Unfortunately we are not able to compute explicitly $\mathcal{V}(\mathcal{D}_r)$ (unless $r = 2$, since $\mathcal{D}_2$ is actually B3x) because our algorithms can only deal with *real* arrangement. We'll return on this example.

Definition 3.39. Let $n \geqslant 2$ be an integer. The *square arrangement* $\mathcal{Q}(n)$ is the arrangement of $2n + 2$ lines in $\mathbb{P}^2(\mathbb{R})$ defined as follows: take n vertical lines and n horizontal lines, arrange them in a grid of $(n-1) \times (n-1)$ squares, and add the two diagonals.

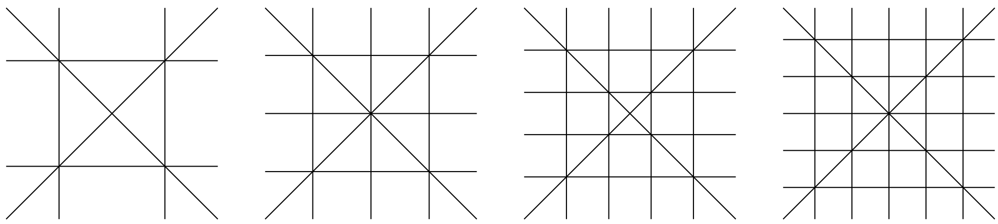


Figure 3.6: Arrangements $\mathcal{Q}(2)$, $\mathcal{Q}(3)$, $\mathcal{Q}(4)$ and $\mathcal{Q}(5)$.

Notice that $\mathcal{Q}(2)$ and A3 are the same arrangement, and so are $\mathcal{Q}(3)$ and B3x. Our computation shows that $\mathcal{V}(\mathcal{Q}(4))$ has one essential 1-dimensional translated component (see Chapter 5 for details).

While searching for interesting arrangements, we came across Grünbaum's catalogue [28] and we decided to focus at first on regular arrangements.

Definition 3.40 ([29])**.** Let $m \geqslant 3$ be an integer. The *regular arrangement* $\mathcal{R}(2m)$ is the arrangement of $2m$ lines in $\mathbb{P}^2(\mathbb{R})$ defined as follows: take the m sides of a regular m-agon, together with the m axes of symmetry.

So, for example, $\mathcal{R}(6)$ is the same of A3, and $\mathcal{R}(8)$ is another name for B3x. The next one is $\mathcal{R}(10)$, and it seems that no one computed its characteristic variety. We took the challenge and used the algorithm `variety_from_matrix` (Section 4.2). It took more or less two weeks to finish, on a computer with an AMD A8-3850 APU @ 2.9 GHz processor, and the result can be seen in Chapter 5.

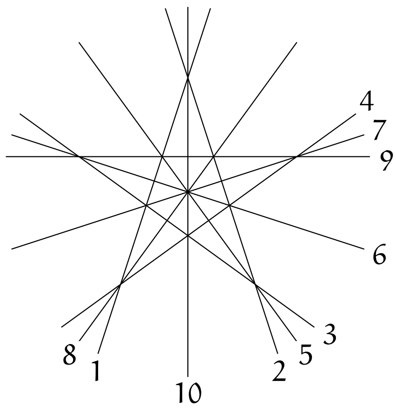


Figure 3.7: The $\mathcal{R}(10)$ arrangement.

Once again, we looked at the result. $\mathcal{V}(\mathcal{R}(10))$ has four translated 0-dimensional essential components. On the other hand, $\mathcal{R}(10)$ has one essential neighbourly partition

$$\pi = \{\{1,6\},\{2,7\},\{3,8\},\{4,5\},\{9,10\}\}$$

which is the double points partition. A small computation reveals that $\mathcal{I}(\pi)$ is 0-dimensional, so it does not contribute to the characteristic variety...

...or does it? Let's have a closer look. If we compute the primary decomposition of (the radical of) $\mathcal{I}(\pi)$, we find seven ideals:

$$\begin{aligned}
I_1 = (&t_7 - t_9,\ t_6 - 1,\ t_4 + t_5 + t_8 + t_9 + 1,\ t_3 - t_5,\ t_2 - t_8,\ t_1 - 1,\ t_9^2 - t_8,\\
&t_8t_9 + t_5 + t_8 + t_9 + 1,\ t_5t_9 - 1,\ t_8^2 - t_5,\ t_5t_8 - t_9,\ t_5^2 + t_5 + t_8 + t_9 + 1),\\
I_2 = (&t_7 - t_8,\ t_6 - t_8,\ t_5 - t_8,\ t_4 - t_9,\ t_3 - t_9,\ t_2 - t_9,\ t_1 - t_9,\\
&t_8t_9 + t_9^2 + t_8 + t_9 + 1,\ t_8^2 - t_9,\ t_9^3 - t_8),\\
I_3 = (&t_7 - 1,\ t_6 - t_9,\ t_4 - t_8,\ t_3 + t_5 + t_8 + t_9 + 1,\ t_2 - 1,\ t_1 - t_5,\ t_9^2 - t_5,\\
&t_8t_9 - 1, t_5t_9 + t_5 + t_8 + t_9 + 1,\ t_8^2 + t_5 + t_8 + t_9 + 1,\ t_5t_8 - t_9,\ t_5^2 - t_8),\\
I_4 = (&t_8 - t_9,\ t_5 - 1,\ t_4 - 1,\ t_3 - t_6,\ t_2 + t_6 + t_7 + t_9 + 1,\ t_1 - t_7,\ t_9^2 - t_6,\\
&t_7t_9 - 1,\ t_6t_9 + t_6 + t_7 + t_9 + 1,\ t_7^2 + t_6 + t_7 + t_9 + 1,\ t_6t_7 - t_9,\ t_6^2 - t_7),
\end{aligned}$$

$$\begin{aligned}
I_5 = (&t_8 - 1,\ t_5 - t_9,\ t_4 - t_7,\ t_3 - 1,\ t_2 - t_6,\ t_1 + t_6 + t_7 + t_9 + 1,\ t_9^2 - t_7,\\
&t_7t_9 + t_6 + t_7 + t_9 + 1,\ t_6t_9 - 1,\ t_7^2 - t_6,\ t_6t_7 - t_9,\ t_6^2 + t_6 + t_7 + t_9 + 1),\\
I_6 = (&t_9 - 1,\ t_5 + t_6 + t_7 + t_8 + 1,\ t_4 - t_6,\ t_3 - t_7,\ t_2 + t_6 + t_7 + t_8 + 1,\\
&t_1 - t_8,\ t_8^2 - t_7,\ t_7t_8 - t_6,\ t_6t_8 + t_6 + t_7 + t_8 + 1,\\
&t_7^2 + t_6 + t_7 + t_8 + 1,\ t_6t_7 - 1,\ t_6^2 - t_8),\\
I_7 = (&t_9 - 1,\ t_8 - 1,\ t_7 - 1,\ t_6 - 1,\ t_5 - 1,\ t_4 - 1,\ t_3 - 1,\ t_2 - 1,\ t_1 - 1).
\end{aligned}$$

Does it look familiar? Yes, *the ideal* I_2 *is the ideal whose zero locus* $\mathcal{Z}(I_2)$ *is exactly the four translated points of* $\mathcal{V}(\mathcal{R}(10))$. It seems that, after all, this neighbourly partition does record the translated components.

Let's look again at B3x and compute $J = \mathcal{I}(\{\{1,4,5\},\{2,3,7\},\{6\},\{8\}\})$, the ideal associated with the double points partition, despite the fact that that partition is *not* neighbourly. It turns out that $\dim(J) = 1$, and the primary decomposition of $\sqrt{J}$ is made of two ideals:

$$\begin{aligned}
I_1 = (&t_6 + 1,\ t_2 - t_3,\ t_1 - t_4,\ t_5t_7 - 1,\ t_4t_7 + t_3,\ t_3t_5 + t_4,\\
&t_4^2 - t_5,\ t_3t_4 + 1,\ t_3^2 - t_7),\\
I_2 = (&t_6 - 1,\ t_2 - t_3,\ t_1 - t_4,\ t_5t_7 - 1,\ t_4t_7 - t_3,\ t_3t_5 - t_4,\\
&t_4^2 - t_5,\ t_3t_4 - 1,\ t_3^2 - t_7).
\end{aligned}$$

The first one gives the translated component! We are tempted to state a conjecture:

> The essential translated components of the characteristic variety $\mathcal{V}(\mathcal{A})$ are found among the zero locus of the ideals in the primary decomposition of $\mathcal{I}(\pi)$, where π is the double points partition of $\mathcal{A}$.

Unfortunately, this conjecture is *false*. While we were writing this work, we managed to compute other characteristic varieties thanks to the improved algorithm `variety_from_matrix_conditioned` (Section 4.3). It turns out that $\mathcal{A}(11,1)$ is a counterexample: its two essential translated components are 0-dimensional, but the ideals of the primary decomposition of $\mathcal{I}(\pi)$ (π is the double points partition) are all 1-dimensional. However, those two point *do* belong to $\mathcal{Z}(\mathcal{I}(\pi))$.

The situation is worse than we expected: after we tried to formulate other hypotheses involving the relationship between translated components and neighbourly partitions, we decided to test if the essential translated components of $\mathcal{V}(\mathcal{A}(11,1))$ appear in the primary decompositions of ideals $\mathcal{I}(\pi)$, where π is *any* partition of the set $\{1,\ldots,11\}$. The result is disappointing—*none* of the primary decompositions of the $\mathcal{I}(\pi)$'s contains the ideal of the two essential translated points of $\mathcal{V}(\mathcal{A}(11,1))$. It seems that neighbourly partitions alone are not sufficient to detect translated components.

Arrangement	Double points partition	Gives translated component?
B3x	$\{\{1,4,5\},\{2,3,7\},\{6\},\{8\}\}$	Yes
$\mathcal{R}(10)$	$\{\{1,6\},\{2,7\},\{3,8\},\{4,5\},\{9,10\}\}$	Yes
$\mathcal{A}(10,2)$	$\{\{1,2,3,6,7,8,10\},\{4\},\{5\},\{9\}\}$	Yes
$\mathcal{Q}(4)$	$\{\{1,10\},\{2,5,6,9\},\{3,4,7,8\}\}$	Yes
$\mathcal{A}(11,1)$	$\{\{1\},\{2,4,5,6,7,8,9,10\},\{3\},\{11\}\}$	No
$\mathcal{R}(12)$	$\{\{1,4,7\},\{2,3,9\},\{5,11,12\},\{6\},\{8\},\{10\}\}$	Yes
$\mathcal{A}(12,2)$	$\{\{1,4,5,6,8,10,11\},\{2\},\{3\},\{7\},\{9\},\{12\}\}$	Yes
$\mathcal{R}(14)$	$\{\{1,8\},\{2,9\},\{3,10\},\{4,7\},\{5,14\},\{6,13\},\{11,12\}\}$	Yes

Table 3.1: Double points partitions of some arrangements in Chapter 5.

However, this behaviour is strange. In *all* the other cases that we computed, the essential translated components are found among the ideals in the primary decomposition of $\mathcal{I}(\pi)$, where π is the double point partition. In Table 3.1 we report the double points partitions of the arrangements in the catalogue of Chapter 5 that have essential translated components.

Despite being able to compute characteristic varieties only for complexified real arrangements, we used this method also on arrangements defined directly over $\mathbb{C}$. The only example that we know comes from [7], where Dan Cohen defines a class of arrangements that have 1-dimensional translated components.

Definition 3.41. Let $r \geqslant 2$ be an integer. The r-th *Cohen arrangement* $\mathrm{Cohen}(r)$[▸1] is the arrangement in $\mathbb{P}^2(\mathbb{C})$ of $3r+2$ lines defined by the following polynomial:

$$Q_{\mathrm{Cohen}(r)}(X_1, X_2, X_3) = X_1 X_2 (X_1^r - X_2^r)(X_1^r - X_3^r)(X_2^r - X_3^r).$$

Let ζ be a primitive r-th root of unity. We label the $3r+2$ lines in $\mathrm{Cohen}(r)$ with the numbers $1, \dots, 3r+2$ in the following way:

- if $i = k$ with $1 \leqslant k \leqslant r$, then $\ell_i = \{X_1 - \zeta^k X_2 = 0\}$;
- if $i = r + k$ with $1 \leqslant k \leqslant r$, then $\ell_i = \{X_1 - \zeta^k X_3 = 0\}$;
- if $i = 2r + k$ with $1 \leqslant k \leqslant r$, then $\ell_i = \{X_2 - \zeta^k X_3 = 0\}$;
- $\ell_{3r+1} = \{X_1 = 0\}$ and $\ell_{3r+2} = \{X_2 = 0\}$.

In [7], Cohen proves that $\mathrm{Cohen}(r)$ has $r-1$ essential 1-dimensional translated components, which are given by

$$C_q := \{(\zeta^q, \dots, \zeta^q, \nu, \dots, \nu, u, \dots, u, u^r, \nu^r) \in (\mathbb{C}^*)^{3r+2} \mid u\nu\zeta^q = 1\}$$

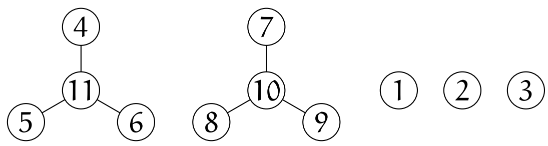
▸1 This is the arrangement that Cohen denotes with $\mathcal{D}_r$ in [7], and that we cited before.

for $q = 1, \dots, r-1$.

Notice that Cohen(2) is actually B3x, so we studied the case $r = 3$. The singular points of Cohen(3) are

$$\begin{aligned} L_2(\text{Cohen}(3)) = \{&\{4,11\},\{5,11\},\{6,11\},\{7,10\},\{8,10\},\{9,10\},\{1,4,9\},\{1,5,7\},\\ &\{1,6,8\},\{2,4,8\},\{2,5,9\},\{2,6,7\},\{3,4,7\},\{3,5,8\},\{3,6,9\},\\ &\{4,5,6,10\},\{7,8,9,11\},\{1,2,3,10,11\}\}, \end{aligned}$$

so the double points graph is



and the double points partition is $\pi = \{\{1\},\{2\},\{3\},\{4,5,6,11\},\{7,8,9,10\}\}$. Let us compute the primary decomposition of $\sqrt{\mathcal{J}(\pi)}$:

$$\begin{aligned}
I_1 = (&t_7+t_8+t_9,\ t_4+t_5+t_6,\ t_2+t_3+1,\ t_1-1,\ t_{10}t_{11}-1,\ t_9^2+t_5t_{10}+t_6t_{10},\\
&t_8t_9-t_5t_{10},\ t_6t_9+t_3+1,\ t_5t_9-t_3,\ t_3t_9-t_8,\ t_8^2-t_6t_{10},\ t_6t_8-1,\\
&t_5t_8+t_3+1,\ t_3t_8+t_8+t_9,\ t_6^2-t_8t_{11},\ t_5t_6-t_9t_{11},\ t_3t_6+t_5+t_6,\\
&t_5^2+t_8t_{11}+t_9t_{11},\ t_3t_5-t_6,\ t_3^2+t_3+1),\\
I_2 = (&t_8-t_9,\ t_7-t_9,\ t_5-t_6,\ t_4-t_6,\ t_2-t_3,\ t_1-t_3,\ t_{10}t_{11}-1,\ t_6t_9+t_3+1,\\
&t_3^2+t_3+1,\ t_9^2t_{11}-t_3t_6,\ t_3t_9t_{11}-t_6^2,\ t_6^2t_{10}-t_3t_9,\ t_3t_6t_{10}-t_9^2,\ t_9^3-t_{10},\\
&t_3t_9^2+t_9^2+t_6t_{10},\ t_6^3-t_{11},\ t_3t_6^2+t_6^2+t_9t_{11}),\\
I_3 = (&t_7+t_8+t_9,\ t_4+t_5+t_6,\ t_2-1,\ t_1+t_3+1,\ t_{10}t_{11}-1,\ t_9^2-t_5t_{10},\\
&t_8t_9-t_6t_{10},\ t_6t_9+t_3+1,\ t_5t_9-1,\ t_3t_9+t_8+t_9,\ t_8^2+t_5t_{10}+t_6t_{10},\\
&t_6t_8-t_3,\ t_5t_8+t_3+1,\ t_3t_8-t_9,\ t_6^2+t_8t_{11}+t_9t_{11},\ t_5t_6-t_8t_{11},\ t_3t_6-t_5,\\
&t_5^2-t_9t_{11},\ t_3t_5+t_5+t_6,\ t_3^2+t_3+1),\\
I_4 = (&t_7+t_8+t_9,\ t_4+t_5+t_6,\ t_3-1,\ t_1+t_2+1,\ t_{10}t_{11}-1,\ t_9^2-t_6t_{10},\\
&t_8t_9+t_5t_{10}+t_6t_{10},\ t_6t_9-1,\ t_5t_9+t_2+1,\ t_2t_9-t_8,\ t_8^2-t_5t_{10},\ t_6t_8-t_2,\\
&t_5t_8-1,\ t_2t_8+t_8+t_9,\ t_6^2-t_9t_{11},\ t_5t_6+t_8t_{11}+t_9t_{11},\ t_2t_6+t_5+t_6,\\
&t_5^2-t_8t_{11},\ t_2t_5-t_6,\ t_2^2+t_2+1),\\
I_5 = (&t_8-t_9,\ t_7-t_9,\ t_5-t_6,\ t_4-t_6,\ t_3-1,\ t_2-1,\ t_1-1,\ t_{10}t_{11}-1,\\
&t_9^2-t_6t_{10},\ t_6t_9-1,\ t_6^2-t_9t_{11}).
\end{aligned}$$

This is the primary decomposition over the "computational" field, i.e. $\mathbb{Q}$; but if we consider the above ideals to belong to the polynomial ring with complex coefficients, we find out that $I_2 = I_2' \cap I_2''$, where

$$I_2' = (t_8 - t_9,\ t_7 - t_9,\ t_5 - t_6,\ t_4 - t_6,\ t_3 - \zeta,\ t_2 - \zeta,\ t_1 - \zeta,\ t_{10}t_{11} - 1,\\ t_9^2 - \zeta t_6 t_{10},\ t_6 t_9 - \zeta^2,\ t_6^2 - \zeta t_9 t_{11}),$$
$$I_2'' = (t_8 - t_9,\ t_7 - t_9,\ t_5 - t_6,\ t_4 - t_6,\ t_3 - \zeta^2,\ t_2 - \zeta^2,\ t_1 - \zeta^2,\ t_{10}t_{11} - 1,\\ t_9^2 - \zeta^2 t_6 t_{10},\ t_6 t_9 - \zeta,\ t_6^2 - \zeta^2 t_9 t_{11}),$$

and ζ is a 3-rd root of unity. It is not hard to see that

$$C_1 = \mathcal{Z}(I_2') \quad \text{and} \quad C_2 = \mathcal{Z}(I_2''),$$

that is to say, also for Cohen(3) it is true that the double points partition encodes the information of at least some of the translated components of its characteristic variety.

3.8 Further Analysis of Regular Arrangements

We focus here on the regular arrangements $\mathcal{R}(2m)$ *with* $m \geqslant 3$ *odd*, trying to deduce some result at least for this class of arrangement.

The properties of the regular arrangements are better understood if we use a particular labelling of the lines, which we call *cyclic labelling*: we number the sides of the m-agon from 0 to $m-1$ cyclically, and use labels $0', \dots, (m-1)'$ for the axes, such that axis i' is the one perpendicular to side i.

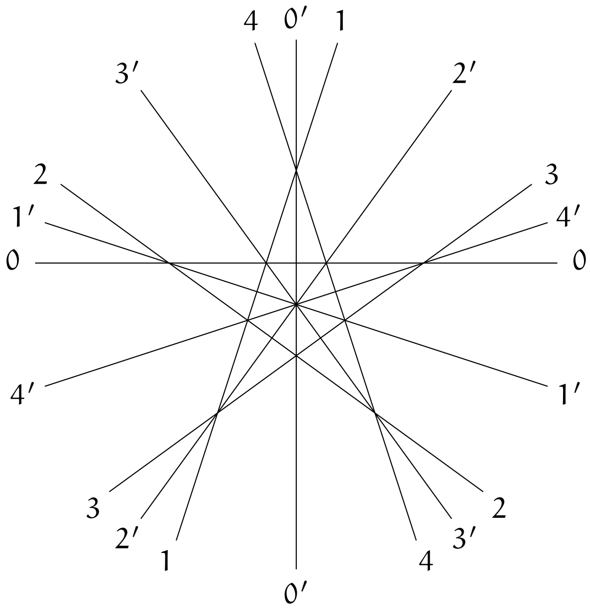


Figure 3.8: Example of cyclic labelling on $\mathcal{R}(10)$.

With the cyclic labelling, it is easy to characterize the elements of $L_2(\mathcal{R}(2m))$:

- there are m double points, which are $\{i, i'\}$ for $i = 0, \dots, m-1$;
- the triple points are $\binom{m}{2}$: for each $i, j \in \{0, \dots, m-1\}$ with $i \neq j$ there is the triple point
$$\left\{ i, j, \left(\frac{i+j}{2} \pmod{m} \right)' \right\}$$
(notice that 2 is always invertible since m is odd);
- the central m-tuple point is $\{0', \dots, (m-1)'\}$.

Lemma 3.42. *The double points partition $\pi = \{\{i, i'\} \mid i = 0, \dots, m-1\}$ is neighbourly.*

Proof. Each double point belongs to a single block of π by definition. For the m-tuple point P, notice that $P \cap \{i, i'\} = \{i'\}$, so the premise of the neighbourly condition (3.3) is always false ($1 \geqslant m-1$).

In general, it is easy to see that, if $\{x, y, z\}$ is a triple point and σ is a neighbourly partition, then x, y and z belong either to the same block, or to three different blocks. In this case a triple point can't be contained in a single block, because all blocks have cardinality 2; the neighbourly condition fails only if
$$\frac{i+j}{2} \equiv i \quad \text{or} \quad \frac{i+j}{2} \equiv j \pmod{m}.$$
Both equivalences imply $i \equiv j \pmod{m}$, which is false by definition of the triple point. □

Lemma 3.43. *The double points partition π is the only (non-trivial) essential neighbourly partition if and only if m is a prime number.*

Proof. $\boxed{\Rightarrow}$ Suppose that $m = pq$ with $p \geqslant 3$, $q \geqslant 3$ odd numbers (not necessarily prime). Then the following partition is neighbourly:
$$\pi' \coloneqq \{\pi_a \mid a = 0, \dots, q-1\}$$
where
$$\pi_a \coloneqq \{a + kq \mid k = 0, \dots, p-1\} \cup \{(a + kq)' \mid k = 0, \dots, p-1\}.$$
Let's prove that π' is neighbourly. The double points are contained in a single block by definition; for the m-uple point P we have
$$P \cap \pi_a = \{(a + kq)' \mid k = 0, \dots, p-1\}$$
therefore $\#(P \cap \pi_a) = p$ and the condition $\#(P \cap \pi_a) \geqslant m-1$ is always false. Now we check the triple points: let $\{i, j, ((i+j)/2)'\}$ be one of them.

- If i and j belong to the same block, i.e. $i = a + k_1 q$ and $j = a + k_2 q$ for some k_1, k_2, then
$$\frac{i+j}{2} \equiv a + \left(\frac{k_1 + k_2}{2} \pmod p\right) q \pmod m$$
and the triple point is contained in a single block of π'.

- If i and j belong to different blocks, i.e. $i = a + k_1 q$ and $j = b + k_2 q$ for some k_1, k_2 and $a \neq b$, then
$$\frac{i+j}{2} \equiv \left(\frac{a+b}{2} \pmod q\right) + \left(\frac{h + k_1 + k_2}{2} \pmod p\right) q \pmod m$$
where h is such that $a + b = ((a+b) \pmod q) + hq$. In particular $((i+j)/2)'$ belongs neither to π_a nor to π_b.

$\boxed{\Leftarrow}$ Let π' be a neighbourly partition. Recall that π' has to be a superpartition of π; suppose that there is a block $\tilde{\pi}$ of π' containing the lines i and j with $i \neq j$ (and lines i', j' too). We want to show that that block must contain also all the other lines.

Since there is a triple point $\{i, j, ((i+j)/2)'\}$, we have that $(i+j)/2 \pmod m$ (and $((i+j)/2)' \pmod m$) belongs to $\tilde{\pi}$. We can repeat the reasoning with the triple points given by $i \cap (i+j)/2$ and $j \cap (i+j)/2$ and obtain that the two values
$$\frac{3i+j}{4}, \frac{i+3j}{4} \pmod m$$
and their primed versions must belong to $\tilde{\pi}$. If we iterate h times, we obtain that all the 2^{h-1} values
$$\frac{(2^h-1)i+j}{2^h}, \frac{(2^h-3)i+3j}{2^h}, \ldots, \frac{i+(2^h-1)j}{2^h} \pmod m$$
must belong to $\tilde{\pi}$. We have the thesis if we prove that for all $r = 0, \ldots, m-1$ there exist $h \geqslant 0$ and $1 \leqslant a \leqslant 2^h - 1$ odd such that
$$r \equiv \frac{ai + (2^h - a)j}{2^h} \pmod m.$$
A bit of elementary algebra brings to
$$a \equiv 2^h \frac{r-j}{i-j} \pmod m \tag{3.8}$$
which is well-defined since $i - j \neq 0$, hence invertible (m is a prime number by hypothesis). Now write
$$\frac{r-j}{i-j} \pmod m = 2^s \tilde{r}$$
(as integers) with $s \geqslant 0$ and $\tilde{r}$ odd; a solution of (3.8) is given by
$$a = \tilde{r}, \qquad h = u(m-1) - s$$
with u big enough so that $a \leqslant 2^h - 1$. □

We will study the regular arrangements $\mathcal{R}(2m)$ with m prime. Let π be the double points partition; we consider $\mathcal{I}(\pi)$ as an ideal in the Laurent polynomial ring

$$\mathbb{C}[T_0^{\pm 1}, \dots, T_{m-1}^{\pm 1}, S_0^{\pm 1}, \dots, S_{m-1}^{\pm 1}]$$

where T_i is the variable associated with line i and S_i is associated with i'. Since the singular points not contained in a single block of π are all the triple points and the m-tuple point, the generators of $\mathcal{I}(\pi)$ are:

- $p(\mathbf{T}, \mathbf{S}) \coloneqq T_0 \cdots T_{m-1} S_0 \cdots S_{m-1} - 1$;
- $q_m(\mathbf{T}, \mathbf{S}) \coloneqq S_0 \cdots S_{m-1} - 1$;
- for all $i, j \in \{0, \dots, m-1\}$ with $i \neq j$, $g_{ij}(\mathbf{T}, \mathbf{S}) \coloneqq T_i T_j S_k - 1$ where $k = (i+j)/2 \pmod{m}$.

Let ω be a primitive m-th root of unity; the point

$$(t_0, \dots, t_{m-1}, s_0, \dots, s_{m-1}) \in (\mathbb{C}^*)^{2m} \tag{3.9}$$

with $t_i = \omega$ and $s_i = \omega^{m-2}$, for $i = 0, \dots, m-1$, belongs to $\mathcal{Z}(\mathcal{I}(\pi))$ (easy verification). In particular we find $m-1$ such points, when ω varies between all primitive m-th roots of unity.

Conjecture 3.44. *These* $m-1$ *points are essential* 0*-dimensional translated components of* $\mathcal{V}(\mathcal{R}(2m))$ *for all primes* $m \geqslant 3$.

For $m = 5$ and $m = 7$ this is verified by computing directly $\mathcal{V}(\mathcal{R}(10))$ and $\mathcal{V}(\mathcal{R}(14))$ (see Chapter 5). For higher values of m, we are able to compute the matrix $[\partial_2]$ with the algorithm `delta2` (Section 4.1) and evaluate it in the points (3.9); all these points belong to the respective characteristic varieties, since they lower the rank of $[\partial_2]$, but we still have to prove that:

1. they belong to $\mathcal{V}(\mathcal{R}(2m))$ for all primes $m \geqslant 3$;
2. they are isolated points.

Chapter 4

Algorithms for Arrangements

In this chapter we present the algorithms that we use to investigate the properties of hyperplane arrangements. Some of them were originally written by Salvetti using the `Axiom` language; we upgraded them to the `SageMath` language [40], in order to make use of its powerful functionalities.

Remark. We present only the main algorithms that we used. Other procedures that we wrote and that are not found in `SageMath` will be explained, but their code won't be outlined.

4.1 The ∂_2 Matrix

Given an affine arrangement `arr` with n lines, the following algorithm computes a matrix with coefficients in $\mathbb{Q}[t_1, \ldots, t_n]$ that represents the boundary operator ∂_2 (Formula (2.2)).

```
def delta2(arr):
    nh=arr.n_hyperplanes()
    RNG=PolyRingQ(nh)
    t=PolyRingVariables(RNG)
    L_sing=sing_up_lo(arr)
    L=order_sing(L_sing)
    d2=[]
    for pt in L: # remember that pt=[S(P),U(P),L(P)]
        SP=pt[0]
        UP=pt[1]
        ns=len(SP)
        nu=len(UP)
        minSP=min(SP) # first line of Cone(P)
```

```
    maxSP=max(SP) # last line of Cone(P)
    for j in xrange(ns-1):
        row=[0 for k in xrange(nh)]
        minC=SP[j] # first line of chamber C_j
        maxC=SP[j+1] # last line of chamber C_j
        for i in xrange(ns): # 1st sum: iterate on the 1-cells in S(P)
            p=1
            p1=1
            p2=1
            hi=SP[i]
            for k in xrange(hi-1):
                if k+1 in UP:
                    p=p*t[k+1]
            if i>j: # distinguish the two cases for [C_i->hi)
                for k in xrange(j+1,i):
                    p1=p1*t[SP[k]]
            else:
                for k in xrange(j+1,ns):
                    p1=p1*t[SP[k]]
                for k in xrange(i):
                    p1=p1*t[SP[k]]
            for k in xrange(i):
                p2=p2*t[SP[k]]
            row[hi-1]=p*(p1-p2)
        for i in xrange(nu): # 2nd sum: iterate on 1-cells in U(P)...
            hi=UP[i]
            p=1
            p1=1
            p2=1
            p3=1
            p4=1
            if minSP<=hi and hi<=maxSP: # ... that belong to Cone(P)
                for k in xrange(hi-1):
                    if k+1 in UP:
                        p=p*t[k+1]
                for k in xrange(j+1,ns): # lines in U(C) (in S(P) and
                ↪ >=maxC)
                    if SP[k]<hi:
                        p1=p1*t[SP[k]]
                    p3=p3*t[SP[k]]
```

```
53          mi=max([hi,maxC])
54          for k in range(ns): # lines in S(P)
55            if SP[k]<hi: # recall that S(P)=L(C)+U(C)
56              p2=p2*t[SP[k]]
57            if SP[k]<hi or SP[k]>=mi:
58              p4=p4*t[SP[k]]
59          row[hi-1]=p*(p1-p2-p3+p4)
60      d2=d2+[row]
61  result=matrix(d2).transpose()
62  return result
```

Let us analyse the previous code more in detail. The input `arr` is supposed to be a hyperplane arrangement in `SageMath` (i.e. an instance of `HyperplaneArrangementElement`). `PolyRingQ(n)` is a procedure that defines the polynomial ring $\mathbb{Q}[\texttt{t\_1}, \dots, \texttt{t\_n}]$; then `PolyRingVariables` defines the vector `t` such that $\texttt{t[i]} = \texttt{t\_i}$.[1]

The procedure `sing_up_lo` returns a list of the singular points of `arr` together with the information about the lines passing above and below. In particular, `L_sing` contains for every $P \in \mathcal{S}ing(\mathcal{A})$ a triple $[S(P), U(P), L(P)]$ where $S(P)$, $U(P)$ and $L(P)$ have the same meaning of Section 2.3. (Actually, in the algorithm they contain the *labels* of the lines of `arr`, not the lines themselves.) `order_sing` orders the list of singular points with respect to the following: given two lists $[S(P_1), U(P_1), L(P_1)]$ and $[S(P_2), U(P_2), L(P_2)]$, we say that $P_1 > P_2$ if $\max(S(P_1)) > \max(S(P_2))$, or $\max(S(P_1)) = \max(S(P_2))$ and $\min(S(P_1)) \in U(P_2)$.

We begin building the matrix `d2`. Notice that the algorithm actually computes $[\partial_2]^T$ and then transposes it just before returning it. For every point `pt`, we extract the meaningful information (lines 9–14) and add $\#(S(\texttt{pt})) - 1$ rows, one for each 2-cell of the form $(C_j, \texttt{pt})$. Then we populate each row with the corresponding coefficients.

- Lines 19–37 deals with the first summation of Formula (2.2), i.e. the lines passing through `pt`; each addend has the form $(\prod t)(\prod t - \prod t)$, and the three products correspond to lines 24–26, 27–34 and 35–36 respectively.

- Lines 38–59 deals with the second summation of Formula (2.2), i.e. the lines in $\mathrm{Cone}(\texttt{pt})$ passing above `pt`; each addend has the form $(\prod t)(1 - \prod t)(\prod t - \prod t)$, and lines 46–48 compute the first product. The remaining part of the addend is computed distributing $(1 - A)(B - C) = B - C - AB + AC$ and computing each of these separately. More precisely, lines 50–51 compute B (lines in $U(C_j)$ less than `hi`) and line 52 computes C (all lines in $U(C_j)$); lines 55–56 compute AB and lines 57–58 compute AC with a trick:

[1] Recall that in `SageMath`, as in `Python`, the first element of a list `L` is `L[0]`, *not* `L[1]`.

– if hi $>$ maxC, then all the lines in $L(C_j)$ are less than hi, so all the lines in $S(\mathtt{pt})$ have to be included;
– if hi $<$ maxC, then no line in $U(C_j)$ is less than hi, so it suffices to take the lines in $S(\mathtt{pt})$ less than hi for factor A and all the lines greater or equal than maxC (i.e. the lines in $U(C_j)$) for factor C.

4.2 The Characteristic Variety

The following algorithm takes a polynomial matrix mat (that represents a boundary ∂_2), the number of variables nv, and an integer k, and returns the primary decomposition of the (radical of the) ideal generated by all the $k \times k$ minors of mat, that is to say, the components of the characteristic variety $\mathcal{V}_k(\mathtt{arr})$ if $\mathtt{mat} = \mathtt{delta2(arr)}$.

```
def zeroset(mat,k,nv):
    RNG=PolyRingQ(nv)
    zeros=[]
    gen_pt=[randint(0,999) for i in xrange(nv)]
    nr=mat.nrows()
    nc=mat.ncols()
    row_list=sorted([sorted(list(x)) for x in Subsets(xrange(nr),k)])
    col_list=sorted([sorted(list(x)) for x in Subsets(xrange(nc),k)])
    notempty=False
    for rr in row_list:
        if notempty:
            break
        for cc in col_list:
            mink=mat.matrix_from_rows_and_columns(rr,cc)
            dk=mink.determinant()
            if dk==0:
                continue
            dkfactors=[x[0] for x in dk.factor() if x[0] not in RNG.gens()]
            list_id=[[x] for x in dkfactors]
            rr0=rr
            cc0=cc
            notempty=True
            break
    if not notempty:
        return []
    for rr in row_list[row_list.index(rr0):]:
        for cc in col_list:
```

```
28          if rr==rr0 and col_list.index(cc)<=col_list.index(cc0):
29             continue
30          mink=mat.matrix_from_rows_and_columns(rr,cc)
31          dk=mink.determinant()
32          if dk==0:
33             continue
34          dkfactors=[x[0] for x in dk.factor() if x[0] not in RNG.gens()]
35          list_id_aux=[]
36          for idl in list_id:
37             matidl=mat.apply_map(lambda x: RNG.ideal(idl).reduce(x))
38             matnum=matidl.apply_map(lambda x: x(gen_pt))
39             if matnum.rank()<k:
40                zeros=zeros+[idl]
41                continue
42             for g in dkfactors:
43                list_id_aux=list_id_aux+[[sqfree(x) for x in
                   ↪  RNG.ideal(idl+[g]).groebner_basis()]]
44          if list_id_aux==[]:
45             continue
46          list_id_red=reduce_ideal_list([RNG.ideal(idl) for idl in
            ↪  list_id_aux])
47          list_id=[idl.groebner_basis() for idl in list_id_red]
48    zeros=zeros+list_id
49    zeros_red=reduce_ideal_list([RNG.ideal(idl) for idl in zeros])
50    return [RNG.ideal(idl.groebner_basis()) for idl in zeros_red]
```

The algorithm begins with the definition of the ring $\texttt{RNG} = \mathbb{Q}[\texttt{t\_1}, \dots, \texttt{t\_n}]$. The list zeros will store the result. On line 4, gen_pt is a random point of $\mathbb{Q}^n$ that will be used later to speed up the computation of the rank.

After having set up the two lists row_list and col_list, that contain the possible choices of rows and columns for a $k \times k$ minor, the algorithm begins to look for a non-zero minor (lines 9–23). For every possible choice of rr and cc, the algorithm computes the corresponding minor (line 14); if it is not zero, its factors are stored in dkfactors[2] and a list of ideals is built—each factor belongs *a priori* to a different component of the primary decomposition.

After line 23, $(\texttt{rr0}, \texttt{cc0})$ is the first non-zero minor, and list_id contains the singletons of its factors; if all minors are zero (lines 24–25), then for all the points of $(\mathbb{C}^*)^n$ the rank of mat is less than k and the empty list is returned. Otherwise, the

[2] Factors of the form t_i are not considered, because the coefficients are thought to be Laurent polynomials (hence the t_i's should be invertible), but SageMath works with "true" polynomials.

algorithm continues with the other minors (lines 26–27), skipping the ones that it knows to be zero (lines 28–29).

When the algorithm finds another non-zero minor (i.e. it passes the check of lines 32–33), it factors the minor and prepares an auxiliary list. Then, for all the "partial components" `idl` of `list_id`, it tries to compute the rank of `mat` under the conditions given by `idl`. To do so, it reduces the coefficients modulo `idl` (line 37) and evaluates the matrix in a random point (line 38)—it is easier to compute the rank of a matrix with rational coefficients than a matrix with polynomial ones. If the rank of `mat` modulo `idl` is low enough, `idl` is an admissible component: the algorithm adds it to the list `zeros` and continues.

The next step is to upgrade the "partial components" of `list_id` with the factors of the new non-zero minor. In lines 42–43, we add to the auxiliary list the new partial components $(\texttt{idl}, g)$ for each factor g of the minor.[3] Notice that a component `idl` that has been added to `zeros` does *not* reach this stage.

When all the partial components of `list_id` have been examined (after line 43), the algorithm checks if `list_id_aux` is empty: in this case, the new minor does not add new information, and the algorithm proceeds with the next one. Otherwise (lines 46–47), it upgrades the list of partial components with the new ones. The procedure `reduce_ideal_list` receives a list of ideals L and returns another list of ideals $L' = [J \in L \mid \nexists I \in L \text{ s.t. } I \subseteq J]$, i.e. the minimal ideals among the ones of L. Since $I \subseteq J$ implies $\mathcal{Z}(I) \supseteq \mathcal{Z}(J)$, where $\mathcal{Z}(I)$ is the zero locus of I, this operation discards embedded components of the variety.

After line 47, all the minors have been computed. The list `zeros` contains the components found during the execution, except maybe the ones that arise from the last minor. Therefore we just add them (line 48) and reduce the list once more (line 49). The list of components is then returned.

The algorithm `zeroset` computes the characteristic variety by examining all $k \times k$ minors. Their number is *huge*. Recall that the matrix has $n = \#(\texttt{arr})$ rows and $\nu = \sum_{P \in \mathcal{S}ing(\texttt{arr})} (m(P) - 1)$ columns, bringing a total of

$$\binom{n}{k}\binom{\nu}{k}$$

minors. Therefore this algorithm becomes unfeasible when the number of lines is more than eight or nine. Let us try to improve it.

Let $R_1, \ldots, R_n$ be the rows of $[\partial_2]$. Since $\partial_1 \partial_2 = 0$, we have a relation

$$(t_1 - 1)R_1 + \cdots + (t_n - 1)R_n = 0. \tag{4.1}$$

[3] Surprisingly, `sqfree` is not part of `SageMath`. It returns the square-free part of the polynomial given in input.

If $t_1 \neq 1$, we can divide by $t_1 - 1$ and obtain

$$R_1 = -\frac{t_2 - 1}{t_1 - 1} R_2 - \cdots - \frac{t_n - 1}{t_1 - 1} R_n.$$

Let $[\partial_2]_i$ be the matrix $[\partial_2]$ with the i-th row removed. The last equation implies

$$\operatorname{rk}([\partial_2]) < n - 1 \quad \Leftrightarrow \quad \operatorname{rk}([\partial_2]_1) < n - 1$$

so zeroset($[\partial_2]_1$,nv-1,nv) actually returns the components of the characteristic variety *not contained in the zero locus of the ideal* $(t_1 - 1)$.

Now suppose $t_1 = 1$; Equation (4.1) becomes

$$(t_2 - 1)R_2 + \cdots + (t_n - 1)R_n = 0$$

and if $t_2 \neq 1$, we can write R_2 as a combination of the other lines; this implies that

$$\operatorname{rk}([\partial_2](1, t_2, \ldots, t_n)) < n - 1 \quad \Leftrightarrow \quad \operatorname{rk}([\partial_2]_2(1, t_2, \ldots, t_n)) < n - 1$$

and we can call zeroset on $[\partial_2]_2(1, t_2, \ldots, t_n)$. This computes the components *contained in the zero locus of* $(t_1 - 1)$ *but not in the one of* $(t_2 - 1)$.

We can continue inductively assuming $t_1 = \cdots = t_i = 1$ and computing the components contained in $\mathcal{Z}(t_1 - 1, \ldots, t_i - 1)$ but not in $\mathcal{Z}(t_{i+1} - 1)$. Instead of calling zeroset once, we call it n times on matrices with $n - 1$ rows. This brings a little improvement to the algorithm: it can compute characteristic varieties of arrangements with up to ten lines in reasonable time.

```
def variety_from_matrix(mat,nv):
   RNG=PolyRingQ(nv)
   t=PolyRingVariables(RNG)
   nr=mat.nrows()
   nc=mat.ncols()
   result=[]
   quotid=RNG.ideal([])
   for i in xrange(nr):
      matred=mat.delete_rows([i])
      matred=matred.apply_map(lambda x: x(i*[1]+t[i+1:]))
      if matred.rank()<nv-1:
         result=result+[quotid]
         break
      list_i=zeroset(matred,nv-1,nv)
      pi=RNG.ideal(t[i+1]-1)
      list_i.remove(pi)
```

```
17        if quotid!=RNG.ideal([]):
18           list_i=[RNG.ideal((u+quotid).groebner_basis()) for u in list_i]
19        quotid=quotid+pi
20        result=result+list_i
21     if result==[]:
22        return [RNG.ideal([t[i+1]-1 for i in xrange(nv)])]
23     result=[x.radical().primary_decomposition() for x in result]
24     result=reduce_ideal_list([x for sublist in result for x in sublist])
25     return result
```

The previous algorithm is quite linear. The ideal `quotid` is (0) in the beginning, and it is $(t_1 - 1, \ldots, t_i - 1)$ after the i-th `for` cycle (lines 8–22).[4] At every cycle, the i-th row is removed from `mat`, then all coefficients are evaluated at $t_1 = \cdots = t_i = 1$ (line 10). If we reached the limit rank (line 11–13), it is useless to proceed further: all the subsequent ideals that the algorithm finds contain the ideal `quotid`, therefore the components associated with them are contained in $\mathcal{Z}(\texttt{quotid})$, which belongs to the characteristic variety.

The algorithm then calls `zeroset` on the reduced matrix, and stores the result in `list_i`. Notice that

$$(t_i - 1)R_i + \cdots + (t_n - 1)R_n = 0$$

implies that $t_i = 1$ always lowers the rank of $[\partial_2]_i(1, \ldots, 1, t_i, \ldots, t_n)$. This means that `list_i` contains $(t_i - 1)$ and the other components in `list_i` do not contain the polynomial $t_i - 1$; since we assume $t_i \neq 1$ during the i-th cycle, we can safely remove $(t_i - 1)$ (line 16).

The remaining part of the algorithm is simple. We put back the conditions $(t_1 - 1, \ldots, t_{i-1} - 1)$ in the components of `list_i` and upgrade the variables (lines 17–20). If `result` is still empty after the `for` cycle, it means that the components are all contained in the one defined by $(t_1 - 1, \ldots, t_n - 1)$, that is the single point $(1, \ldots, 1)$, and we return that ideal. Otherwise, we compute again the primary decompositions (the output of `zeroset` is a list of primary ideals, but they can no longer be so once we add relations $t_i - 1$), merge the results and reduce again the list.

4.3 The Characteristic Variety, Part II

Despite the improvement, the algorithms outlined in the previous section do not allow us to compute characteristic varieties for arrangement with too many lines. Of course,

[4] In this description, the variable i in the text and the variable `i` in the algorithm have a slight different meaning, because indices in `SageMath` begin with `0`. The relation is $i = \texttt{i} + 1$.

this is due to the amount of minors involved. The next algorithm avoids the problem: it computes the rank of the matrix DD by reducing it in echelon form.

The idea behind the algorithm is the following: if the first entry is a non-zero polynomial p, bifurcate the computation assuming $p = 0$ on one branch and $p \neq 0$ on the other. In the branch $p = 0$, reduce the matrix modulo p and repeat; in the branch $p \neq 0$, row-reduce the matrix in order to have zero entries under p, then call the algorithm on the submatrix obtained by removing the first row and column.

Let us have a closer look at the algorithm. Its inputs are a matrix DD with polynomial coefficients, the number of variables n, the rank k (i.e. we want the matrix to have rank *strictly less than* k), the "known rank" rk (which is 0 in the beginning), and two lists of polynomials: zero contains the polynomials that are assumed to be 0, and nonzero contains the ones that are assumed to be different than 0; both lists are empty in the beginning.

```
def variety_from_matrix_conditioned(DD,n,k,rk=0,zero=[],nonzero=[]):
   R=PolyRingQ(n)
   if rk>=k:
      return []
   if DD.nrows()==0 or DD.is_zero():
      component=R.ideal(zero)
      for p in nonzero:
         component=component.saturation(R.ideal(p))[0] # saturation returns a pair (ideal,saturation index)
      return [R.ideal(component.groebner_basis())]
   D=copy(DD) # just precautionary
   found=False
   minpoly=R.zero()
   i0=0
   j0=0
   for i in xrange(D.nrows()):
      for j in xrange(D.ncols()):
         if D[i,j].numerator()!=R.zero():
            fact=[f[0] for f in remove_multi_t_powers(D[i,j].numerator()).factor()]
            if Set(fact).issubset(Set(nonzero)):
               found=True
               break
            if minpoly.is_zero() or D[i,j].numerator()<minpoly:
               minpoly=D[i,j].numerator()
               i0=i
```

```
                    j0=j
        if found:
            break
    if found:
        if i!=0:
            D.permute_rows(Permutation([(1,i+1)]))
        if j!=0:
            D.permute_columns(Permutation([(1,j+1)]))
        for i in xrange(1,D.nrows()):
            D=D.with_added_multiple_of_row(i,0,-D[i,0]/D[0,0])
        redD=copy(D).apply_map(lambda q: R(q.numerator()).reduce(R.ideal(zero))/q.denominator())
        return variety_from_matrix_conditioned(redD[1:,1:],n,k,rk+1,zero,nonzero)
    else:
        res=[]
        if i0!=0:
            D.permute_rows(Permutation([(1,i0+1)]))
        if j0!=0:
            D.permute_columns(Permutation([(1,j0+1)]))
        fact=[f[0] for f in remove_multi_t_powers(D[0,0].numerator()).factor()]
        for f in fact:
            gb=R.ideal(zero+[f]).groebner_basis()
            avoid=False
            for g in nonzero:
                if g in R.ideal(gb):
                    avoid=True
                    break
            if avoid:
                continue
            newD=copy(D).apply_map(lambda q: R(q.numerator()).reduce(R.ideal(gb))/q.denominator())
            res=res+variety_from_matrix_conditioned(newD,n,k,rk,list(gb),nonzero)
        for i in xrange(1,D.nrows()):
            D=D.with_added_multiple_of_row(i,0,-D[i,0]/D[0,0])
        redD=copy(D).apply_map(lambda q: R(q.numerator()).reduce(R.ideal(zero))/q.denominator())
```

```
58        res=res+variety_from_matrix_conditioned(redD[1:,1:],n,k,rk+1,zero,nonzero+fact)
59        res=reduce_ideal_list(res)
60        res=sum([J.radical().primary_decomposition() for J in res],[])
61        return reduce_ideal_list(res)
```

If the known rank `rk` has already reached the limit rank k, the conditions define an ideal that does not give a component of the characteristic variety, and the branch closes returning nothing (lines 3–4). After line 4, the original matrix `DD` has surely rank strictly less than k, and we can check if the terminating conditions are met: either there are no more rows, or the new matrix `DD` is the zero matrix. In the latter case, it means that the algorithm managed to reduce `DD` to a form like

$$\begin{pmatrix} d_1 & * & * & * & \cdots & * \\ 0 & \ddots & * & * & \cdots & * \\ 0 & 0 & d_h & * & \cdots & * \\ 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix}$$

where $d_1, \dots, d_h$ are non-zero rational functions with $h \leqslant k-1$ and $*$ denotes possible non-zero entries. In any case, this means that $\mathrm{rk}(\mathtt{DD}) = \mathtt{rk} \geqslant k-1$, so the list `zero` defines an ideal whose zero locus belongs to the characteristic variety. The algorithms takes care also of the `nonzero` condition by saturating the ideal $(\mathtt{zero})$ with respect to each $p \in (\mathtt{nonzero})$: the result of this operation is an ideal I such that

$$\mathcal{Z}(I) = \mathcal{Z}(\mathtt{zero}) \cap \bigcap_{p \in \mathtt{nonzero}} \mathcal{Z}(p)^c,$$

and the algorithm returns (a Gröbner basis of) that ideal.

After line 10, we are sure that `D` has not achieved the limit rank, and there is still a non-zero coefficient. We want to put it in the top-left corner of the matrix and use it to reduce the first column in echelon form. The algorithm looks for a non-zero element following two criteria:

1. first of all, it looks for a coefficient whose factors are all in the `nonzero` list, so that this coefficient can be used to reduce the matrix without bifurcate further;

2. meanwhile, it looks for the minimum non-zero coefficient with respect to the term order defined on the ring `R` (which is DEGREVLEX by default), in the hope that subsequent computation of Gröbner bases won't get too big.

The lines 11–27 do this search. The procedure `remove_multi_t_powers` takes a polynomial p and returns a polynomial q such that $T_i \nmid q$ for all $i = 1, \dots, n$ and $p(\mathbf{T}) = T_1^{a_1} \cdots T_n^{a_n} q(\mathbf{T})$ for some $a_1, \dots, a_n \in \mathbb{N}$. This is needed because the algorithm works with "true" polynomials, but we interpret them as Laurent polynomials. After line 27 there are two possible outcomes: either `found` is true and $(\mathtt{i}, \mathtt{j})$ is the position of a *known* non-zero coefficient (i.e. its factors are in `nonzero`), or `found` is false and $(\mathtt{i0}, \mathtt{j0})$ is the position of the minimum non-zero polynomial. In the former case, lines 28–36 are executed; in the latter, lines 37–61 are.

If `found` is true, the algorithm permutes the rows and columns so to have the non-zero polynomial in position $(0, 0)$, then the matrix is reduced (lines 33–34). During this operation, it is possible that some polynomials belonging to $(\mathtt{zero})$ appear among the coefficients. Notice that the algorithm *does not know* that these are assumed to be zero: its working ring is $\mathtt{R}$, not $^{\mathtt{R}}/_{(\mathtt{zero})}$. Therefore we need to get rid of these extra polynomials (line 35), because they may distort the rank computation. The algorithm then calls itself on the matrix without the first row and column, with the new known rank equal to $\mathtt{rk} + 1$.

If `found` is false, we need to bifurcate. The list `res` will contain the result of the computation. After having put the non-zero polynomial in the top-left corner (lines 39–42), the algorithm factors it (line 43) and opens a new branch for each factor `f` (lines 44–54). First of all, the algorithm computes a Gröbner basis of the ideal $(\mathtt{zero}, \mathtt{f})$, then it checks the compatibility of `f` with the `nonzero` conditions: a polynomial $g \in \mathtt{nonzero}$ cannot belong to the ideal $(\mathtt{zero}, \mathtt{f})$, because that ideal contains the polynomials that are assumed to be zero. For example, suppose that $t_1 t_3 - 1 \in \mathtt{zero}$ and $t_3 - 1 \in \mathtt{nonzero}$; a human being interprets this conditions as "t_3 and t_1 are inverse of each other, so if $t_3 \neq 1$ it can't be $t_1 = 1$" and will discard a factor $t_1 - 1$. This is what lines 46–52 do—in the example we have $t_3 - 1 \in (t_1 t_3 - 1, t_1 - 1)$. If the branch is compatible, the algorithm reduces the matrix modulo $(\mathtt{zero}, \mathtt{f})$ and calls itself *without removing rows and/or columns*, *without upgrading the known rank* and upgrading the `zero` list (line 54).

Once all the factors have branched, the algorithm opens the last branch, in which the pivot is supposed non-zero. Lines 55–58 are similar to lines 33–36, only in this case the list `nonzero` is upgraded with the new known non-zero polynomials, i.e. the factors in `fact`. Finally, the results of all branches are collected and reduced; the primary decompositions are computed, and the result is once again reduced and returned.

Despite the numerous branching, this algorithm is *far more faster* than the one presented in the previous section. As an example, compare the computation time needed to compute the characteristic variety of aB3x (see Chapter 5 for the reference) with a processor Intel® Xeon® E5-2643 v4 @ 3.4GHz (Table 4.1).

Unfortunately, `variety_from_matrix_conditioned` has its limits: the problem here is the computation of Gröbner bases. The polynomials in the successive reductions of the matrix can become very big and both the time and the memory needed to compute

Algorithm	time
variety_from_matrix	13 min 10 s
variety_from_matrix_conditioned	17.6 s

Table 4.1: Comparison of times between different algorithms that compute the characteristic variety.

the Gröbner bases grow too much. However this algorithm successfully computes characteristic varieties of arrangements with up to 14 lines with little trouble.

4.4 Neighbourly Partitions

In this section we present a couple of algorithms that we have developed in order to study the relationship between neighbourly partitions and characteristic varieties.

The first algorithm returns the list of non-local neighbourly partitions of all subarrangements of the *affine* arrangement arr. Recall that we need to look for neighbourly partitions only among superpartitions of the double points partition (Definition 3.19).

The algorithm actually returns a list of pairs $(\mathrm{supp}(\pi), \pi)$ where π is a non-local neighbourly partition for the subarrangement of arr defined by the lines in $\mathrm{supp}(\pi)$.

```
def nonlocal_neighbourly_partitions(arr):
   n=arr.n_hyperplanes()
   S=sing_with_infinity(arr)
   neigh=[]
   for subarr in Subsets(n+1):
      Ssub=[[h for h in s if h in subarr] for s in S]
      Ssub=[s for s in Ssub if len(s)>=2]
      if len(Ssub)<=1:
         continue
      G=Graph([list(subarr),[s for s in Ssub if len(s)==2]])
      if G.is_connected():
         continue
      Part=SetPartitions(make_partition(G.connected_components()))
      for setpart in Part:
         if len(setpart)==1:
            continue
         partition=make_partition([sum(p,Set([])) for p in setpart])
         neighbourly=True
         for part in partition:
```

```
20                for pt in Ssub:
21                    if part.intersection(Set(pt)).cardinality()>=len(pt)-1 and
                       ↪ not Set(pt).issubset(part):
22                        neighbourly=False
23                        break
24                if not neighbourly:
25                    break
26            if neighbourly:
27                neigh+=[(subarr,partition)]
28    return neigh
```

The procedure `sing_with_infinity` returns $L_2(\mathtt{arr})$ completed with the singularity at infinity of `arr`, that is to say, if $n = \#(\mathtt{arr})$, then `sing_with_infinity(arr)` includes also the sets $\{i_1, \dots, i_k, n+1\}$ whenever $\ell_{i_1}, \dots, \ell_{i_k}$ are parallel lines of `arr`.[5]

The algorithm passes through all possible subarrangements of `parr` (line 5) and accumulates the result in the list `neigh`. For each subarrangement `subarr`, the algorithm builds the set of singular points $L_2(\mathtt{subarr})$ (lines 6–7); then it checks if the subarrangement is trivial (lines 8–9): the condition `len(Ssub)` $\leqslant 1$ is satisfied if and only if either the arrangement is central (only one singular point), or it has only one line (no singular points at all), or it is empty. These trivial cases are discarded.

If `subarr` is not trivial, its double points graph is built (line 10); if it is connected, the only neighbourly partition of `subarr` is the trivial one, and the algorithm continues with the next subarrangement; otherwise, the algorithm defines the set of partitions of the set whose elements are the blocks of the double point partition (line 13). By merging those blocks we obtain the superpartitions of the double points partition. For example, suppose that the double points partition is $\{\{1,2\},\{3\},\{4\}\}$. Then `Part` has five elements:

$$\{\{\{1,2\},\{3\},\{4\}\}\}, \quad \{\{\{1,2\}\},\{\{3\},\{4\}\}\}, \quad \{\{\{1,2\},\{3\}\},\{\{4\}\}\},$$
$$\{\{\{1,2\},\{4\}\},\{\{3\}\}\}, \quad \{\{\{1,2\}\},\{\{3\}\},\{\{4\}\}\},$$

and if we remove the innermost braces we get the superpartitions

$$\{\{1,2,3,4\}\}, \quad \{\{1,2\},\{3,4\}\}, \quad \{\{1,2,3\},\{4\}\},$$
$$\{\{1,2,4\},\{3\}\}, \quad \{\{1,2\},\{3\},\{4\}\}.$$

Here `make_partition` is an auxiliary function that takes a list of lists of numbers with empty intersections and returns the same list as object of type `Partition`.

For each partition in `Part`, first of all the algorithm checks if it induces the trivial neighbourly partition (lines 15–16)—in that case, it continues with the next element of `Part`. Then it recovers the superpartition (line 17) and verifies the neighbourly

[5] For an affine arrangement arr with n lines, the infinity line is always denoted by $n+1$.

condition (lines 18–25). If the partition is neighbourly, the algorithm adds it to the result (line 27).

The next algorithm computes the ideal associated with a partition `partition`. Its input are the affine arrangement `arr` of n lines and a partition `partition` with $\mathrm{supp}(\texttt{partition}) \subseteq [n+1]$.

```
1  def ideal_from_partition(partition,arr):
2     n=arr.n_hyperplanes()
3     Sing=sing_with_infinity(arr)
4     R=PolyRingQ(n)
5     S=PolyRingQ(n+1)
6     t=PolyRingVariables(S)
7     infty=prod([t[i] for i in xrange(1,n+2)],1)-1
8     ideal=[infty]
9     ground_set=sum(partition,Set([]))
10    for i in xrange(1,n+2):
11       if i not in ground_set:
12          ideal+=[t[i]-1]
13    Ssub=[[h for h in s if h in ground_set] for s in Sing]
14    Ssub=[s for s in Ssub if len(s)>=2]
15    for pt in Ssub:
16       sub=False
17       for p in partition:
18          if Set(pt).issubset(Set(p)):
19             sub=True
20             break
21       if sub:
22          continue
23       poly=1
24       for i in pt:
25          poly*=t[i]
26       ideal+=[poly-1]
27    tot_ideal=S.ideal(ideal)
28    return R.ideal(tot_ideal.elimination_ideal([t[n+1]]))
```

After the usual definitions (lines 2–6), the algorithm defines the polynomial $\texttt{infty} = t_1 \cdots t_{n+1} - 1$ and places it among the generators of the ideal by default (lines 7–8); then computes $\texttt{ground\_set} = \mathrm{supp}(\texttt{partition})$ (line 9). For every $i \notin \mathrm{supp}(\texttt{partition})$, the polynomial $t_i - 1$ is added to the generators of the ideal (lines 10–12; see also Proposition 3.37).

The algorithm computes now the singular points of the subarrangement defined by `ground_set` (lines 13–14). For each of those, the algorithm checks if the point is contained in a single block of `partition`: if that is the case, it continues with the next singular point (lines 16–22). If a point passes this control, the corresponding polynomial is added to the generators of the ideal (lines 23–26).

Finally, the algorithm computes the ideal in $\mathbb{Q}$[`t_1`, ..., `t_{n+1}`] and eliminates the variable `t_{n+1}`, returning an ideal that can be analysed along with the ideals of the characteristic variety of an affine arrangement.

4.5 Wiring Diagrams

One of the difficulties that one encounters while working with arrangements in `SageMath` is that most functions do not work, or are not implemented, if the arrangement is not defined in an exact field like $\mathbb{Q}$. Real numbers are difficult for a computer to deal with. Imagine the situation represented in Figure 4.1: if the lines are approximations, a triple point could span three double points, destroying the combinatorics of the arrangement.

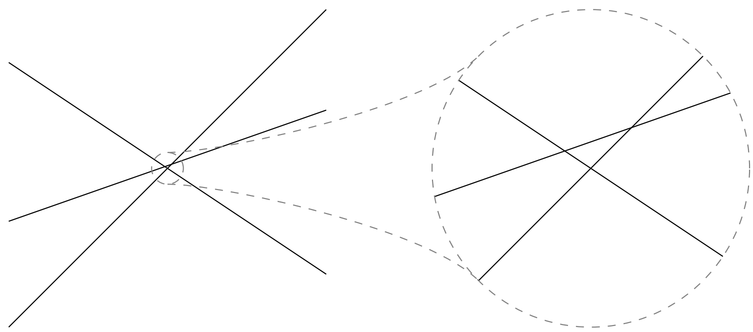


Figure 4.1: Effects of the approximation of real lines.

A possible solution to this problem is to encode the information of an arrangement in another structure which is more easily understandable for a computer. One of such structures is that of wiring diagrams. It is easier to show how to build a wiring diagram from an arrangement than to formally define what a wiring diagram is (see Figure 4.2 for reference).

Given an arrangement $\mathcal{A} = \{\ell_1, \ldots, \ell_n\}$ in $\mathbb{R}^2$, pick a line (*guiding line*) L such that

- no point in $\mathcal{S}ing(\mathcal{A})$ belongs to L;
- the projection map $\mathcal{S}ing(\mathcal{A}) \to L$ is injective, i.e. each line orthogonal to L contains at most one point of $\mathcal{S}ing(\mathcal{A})$.

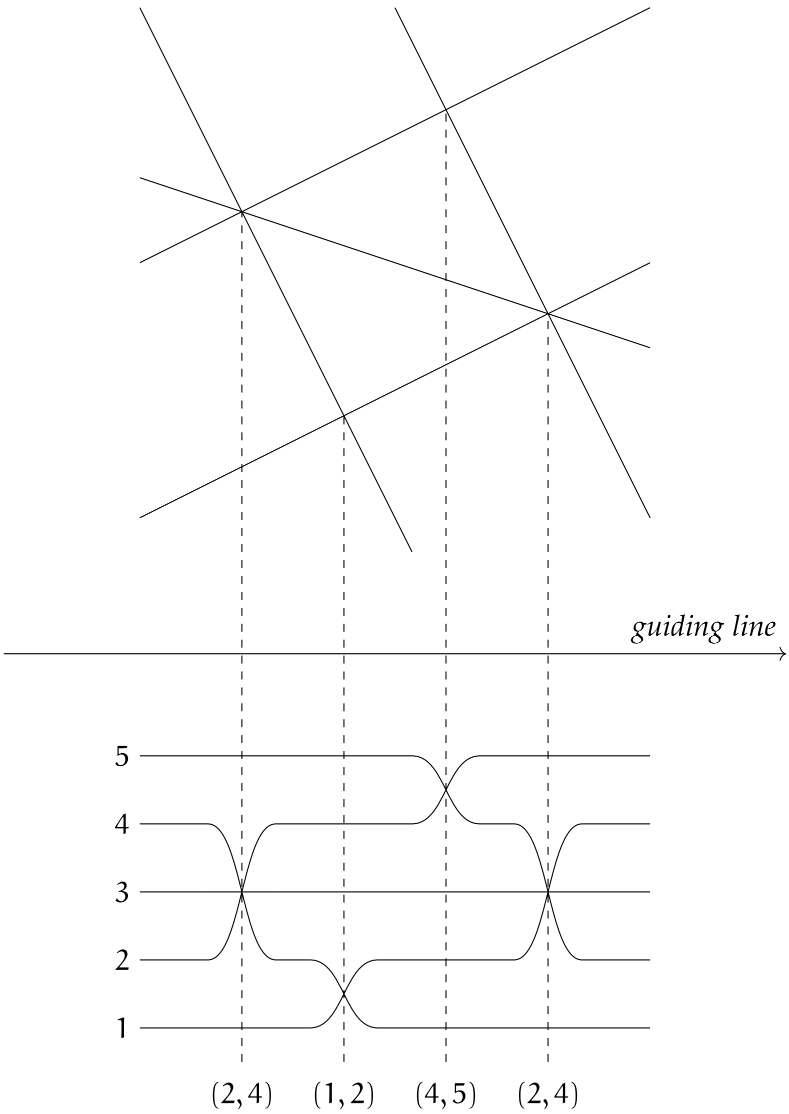


Figure 4.2: Wiring diagram and sequence of Lefschetz pairs associated with the arrangement aA3.

Then begin with n horizontal parallel *wires*, or *strands*, numbered $1, \ldots, n$ from bottom to top, and go through the guiding line. Every time you come across a projection of a point in $\mathcal{S}ing(\mathcal{A})$, make the correspondent switch in the strands.

To a wiring diagram it is possible to associate a list of *Lefschetz pairs*, that is a list of $\#(\mathcal{S}ing(\mathcal{A}))$ pairs (i, j) with $1 \leqslant i < j \leqslant n$. Each pair is associated with an intersection point P in the following way: number *locally* the wires before the point with $1, \ldots, n$ from bottom to top; the corresponding pair is made of the minimum and maximum labels of the wires intersecting at P.

Wiring diagrams are useful when dealing with *pseudolines arrangements*. We recall briefly the definition, leaving the details to [4, Chapter 6].

Definition 4.1. A simple closed curve L embedded in $\mathbb{P}^2(\mathbb{R})$ is a *pseudoline* if $\mathbb{P}^2(\mathbb{R}) \setminus L$ has one connected component.

Lemma 4.2. *Let L_1 and L_2 be two pseudolines. Then $L_1 \cap L_2 \neq \varnothing$. Moreover, if $L_1 \cap L_2$ is a single point P, then L_1 and L_2 intersect transversally.*

Definition 4.3. An *arrangement of pseudolines* is a finite collection of pseudolines $\mathcal{A} = \{L_1, \ldots, L_n\}$ such that $\cap \mathcal{A} = \varnothing$ and for every $L_i, L_j \in \mathcal{A}$ with $i \neq j$ we have $L_i \cap L_j = \{\text{one point}\}$.

In order to define *affine* pseudolines and their arrangements, we have to be more subtle. Consider $\mathbb{P}^2(\mathbb{R})$ as the quotient of the closed 2-disk D^2 by the equivalence relation that identifies antipodal points on ∂D^2, and consider the affine plane $\mathbb{R}^2$ as the interior of D^2.

Definition 4.4. An *affine pseudoline* is the image $L = f((0,1))$ of a continuous injective map $f\colon [0,1] \to D^2$ such that $f^{-1}(\partial D^2) = \{0,1\}$. An *arrangement of affine pseudolines* is a finite collection of affine pseudolines $\mathcal{A} = \{L_1, \ldots, L_n\}$ such that

1. for all $L_i, L_j \in \mathcal{A}$ with $i \neq j$ we have $\#\big(L_i \cap L_j\big) \leqslant 1$;
2. if $\#\big(L_i \cap L_j\big) = 1$, the intersection is transversal;
3. the "pseudo-parallelism" relation
$$L_i \parallel L_j \quad \Leftrightarrow \quad L_i = L_j \text{ or } L_i \cap L_j = \varnothing$$
is an equivalence relation;
4. for any $L_i, L_j \in \mathcal{A}$ such that $L_i \cap L_j = \varnothing$ there exists $L_k \in \mathcal{A}$ such that $L_i \cap L_k \neq \varnothing$ and $L_j \cap L_k \neq \varnothing$.

Obviously a line arrangement is also a pseudoline arrangement. However, the class of pseudoline arrangements has something more.

Definition 4.5. An arrangement of pseudolines is *stretchable* if there is a self-homeomorphism of the projective plane such that the image of each pseudoline of the arrangement is a (straight) line.

There exist non-stretchable pseudoline arrangements, obtained for example by violating theorems of Projective Geometry; in Figure 4.3 we see an example. This is not possible for "small" arrangements.

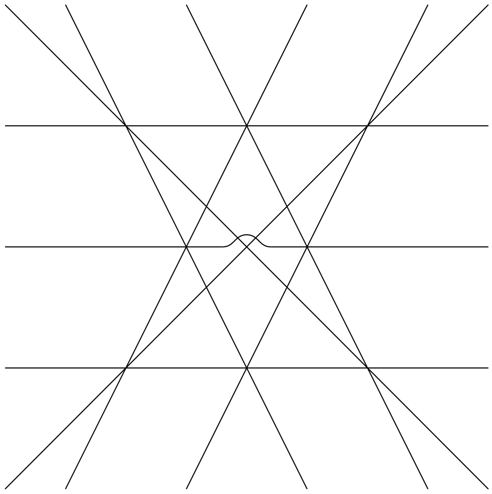


Figure 4.3: This arrangement of pseudolines is non-stretchable, because of Pappus's Theorem.

Theorem 4.6. *All pseudoline arrangements with at most* 8 *lines are stretchable ([27]). All simplicial pseudoline arrangements with at most* 14 *lines are stretchable, and there exist non-stretchable simplicial pseudoline arrangements with* 15 *lines ([17]).*

`SageMath` has a class called `PseudolineArrangement`, but its functionality is quite limited. We decided to implement a class ourselves, whose instances are representations of wiring diagrams.

```
class WiringDiagram():
   def __init__(self,pairs,n_strands):
      try:
         maxindex=max(i for pair in pairs for i in pair)
         minindex=min(i for pair in pairs for i in pair)
      except ValueError:
         maxindex=0
         minindex=1
      if maxindex>n_strands or minindex<=0:
         raise ValueError("permutations out of range")
      self._pairs=pairs
      self._n_strands=n_strands

   def __repr__(self):
      desc="Wiring diagram on "
      if self._n_strands==1:
         desc+="1 strand "
```

```
        else:
            desc+="{} strands ".format(self._n_strands)
        if len(self._pairs)==1:
            desc+="with 1 singularity"
        else:
            desc+="with {} singularities".format(len(self._pairs))
        return desc

    def n_strands(self):
        return self._n_strands

    def pairs(self):
        return self._pairs

    def end_configuration(self):
        n=self._n_strands
        conf=list(xrange(1,n+1))
        for pair in self._pairs: # apply the corresponding permutation
            p=make_inverting_perm(pair,n)
            conf=p.action(conf)
        return tuple(conf)

    def singularities(self,upper_and_lower=False):
        n=self._n_strands
        conf=list(xrange(1,n+1))
        res=[]
        for pair in self._pairs: # extract the three lists...
            minp=min(pair)
            maxp=max(pair)
            LP=conf[:minp-1]
            SP=conf[minp-1:maxp]
            UP=conf[maxp:]
            res.append([SP,UP,LP])
            p=make_inverting_perm(pair,n)
            conf=p.action(conf) # ... then apply the permutation
        if upper_and_lower:
            return res
        else:
            return [s[0] for s in res]

```

```
    def remove_strands(self,strand_list):
        wd=self
        for s in sorted(strand_list,reverse=True): # from the top, so it is
        ↪ not necessary relabeling
            Pairs=wd.pairs()
            Sing=wd.singularities(upper_and_lower=True)
            n=wd.n_strands()
            res=[]
            for j in xrange(len(Pairs)):
                if s in Sing[j][2]: # i.e. s in L(P)
                    res+=[[Pairs[j][0]-1,Pairs[j][1]-1]] # shift below
                elif s in Sing[j][1]: # i.e. s in U(P)
                    res+=[Pairs[j]] # leave as it is
                else: # i.e. s in S(P)
                    temp=[Pairs[j][0],Pairs[j][1]-1]
                    if temp[0]!=temp[1]: # the pair does not die
                        res+=[temp]
            wd=WiringDiagram(res,n-1)
        return wd
```

A wiring diagram is defined by two objects: the list of Lefschetz pairs and the number of strands; while initializing a `WiringDiagram` object, we check that these two data are compatible.

We describe briefly the methods available to a `WiringDiagram` object.

`n_strands()` Return the number of strands.

`pairs()` Return the list of the Lefschetz pairs.

`end_configuration()` Return a tuple with the labels of the strands as they appear on the right of the diagram, ordered from bottom to top. Notice that the strands are always ordered $(1, \dots, n)$ from bottom to top on the left. If `wd` represents a pseudoline arrangement of n lines, `wd.end_configuration()` should be $(n, \dots, 1)$.

`singularities(upper_and_lower=False)` Return the list of the singular points of the diagram. Each point is represented by the list of the lines passing through it. If `upper_and_lower` is true, each singularity P is represented by a list of three elements $[S(P), U(P), L(P)]$ where:

- $S(P)$ is the list of the lines passing through P;
- $U(P)$ is the list of the lines passing above P;
- $L(P)$ is the list of the lines passing below P.

In other words, suppose that P is represented by the Lefschetz pair $[i, j]$ with $i < j$. Then the strands configuration immediately before P can be written (read from bottom to top) as $[\ L(P) \mid S(P) \mid U(P)\ ]$, where $L(P)$ are the strands from the 1st to the $(i-1)$-th, $S(P)$ are the strands from the i-th to the j-th, and $U(P)$ are the strands from the $(j+1)$-th to the last.

`remove_strands(strand_list)` Return a `WiringDiagram` obtained by removing the strands specified in `strand_list`.

In the code for `WiringDiagram`, the auxiliary function `make_inverting_perms` takes as input a pair (i, j) and an integer n and returns the permutation

$$\begin{pmatrix} 1 & \cdots & i-1 & i & i+1 & \cdots & j-1 & j & j+1 & \cdots & n \\ 1 & \cdots & i-1 & j & j-1 & \cdots & i+1 & i & j+1 & \cdots & n \end{pmatrix}.$$

Example 4.1. A realization of $a\mathcal{R}(10)$ as a wiring diagram is pictured in Figure 4.4. Notice that $\mathcal{R}(10)$ is "essentially irrational", that is to say, it is not isomorphic to any arrangement in $\mathbb{P}^2(\mathbb{Q})$ (see [29, p. 33 ff.]).

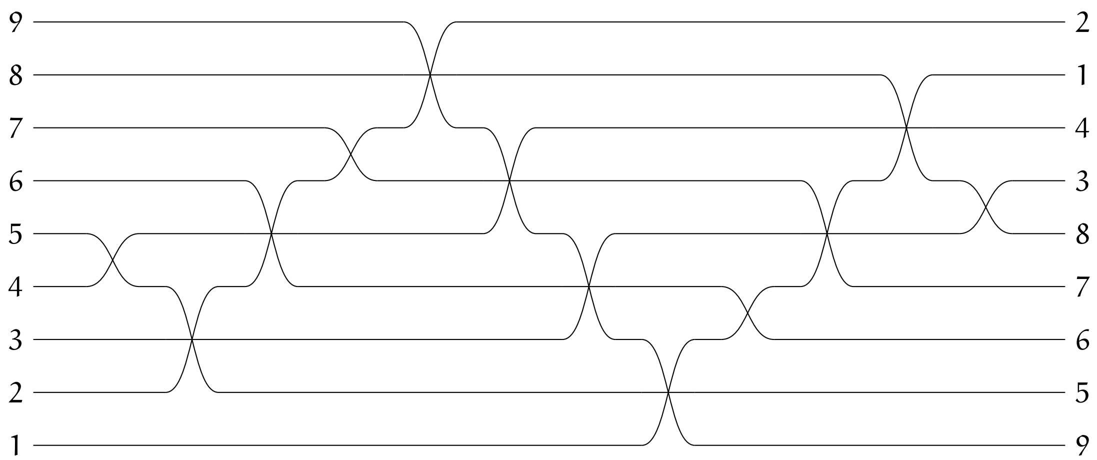


Figure 4.4: A wiring diagram for $a\mathcal{R}(10)$. Each strand corresponds to the line with the same number in Chapter 5, with line 10 sent to infinity.

All the algorithms of the previous sections have been adapted to work with wiring diagrams. In order to avoid being repetitive, we won't report the actual code here.

4.6 Some Remarks on the Algorithms of Sections 4.2 and 4.3

The task of both algorithms of Sections 4.2 and 4.3 is: given a (Laurent) polynomial matrix M and a target rank k, compute the (primary decomposition of the radical of the) ideal generated by all $k \times k$ minors, which describes the set of points P such that the rank of M evaluated at P is strictly less than k. The first algorithm computes one minor at a time, with some tricks that speed up the computation, while the second one computes directly the rank through a series of bifurcations.

While both algorithms can be used to compute characteristic varieties, the second one appears to be far more efficient: in fact, the huge number of minors even for relatively small matrices is what hinders the first algorithm, that becomes too slow for arrangements with more than 10 lines. The second algorithm does not suffer from this problem, but it uses Gröbner bases, and experiments show that due to this it begins to struggle with arrangements with 14-15 lines.

Another problem comes from the intrinsic nature of the computer: it can deal only with *exact* numbers, i.e. rational. Therefore at first we were limited to arrangements whose lines admit equations with rational coefficients. Subsequently we adapted our algorithms to wiring diagrams, which are semi-combinatorial descriptions of line arrangements, so that we can now compute characteristic varieties for more general arrangements, among which "essentially irrational" ones (see Theorem 2.25 of [29]).

Despite the fact that these algorithms were designed with a specific matrix in mind, that is the ∂_2 matrix of the algebraic complex defined by Salvetti and Settepanella [43] and refined by Gaiffi and Salvetti [26] (see Section 2.3), they can be applied to *any* matrix with polynomial coefficients, in particular to the Alexander matrix (Definition 3.5). This allows us to deal not only with complexified real arrangements (for which the above algebraic complex is defined), but also with complex arrangements, as long as we are able to compute a presentation of the fundamental group of the complement.

Chapter 5

Catalogue of Remarkable Projective Line Arrangements

In this chapter we present a compendium of some projective line arrangements. Some of them can be found in the literature and have been studied in deep (although their names seem to change between different articles); some other appear in the literature, but their characteristic varieties have not been computed as far as we know. Most of the entries come from Grümbaum's catalogue of simplicial arrangements [28].

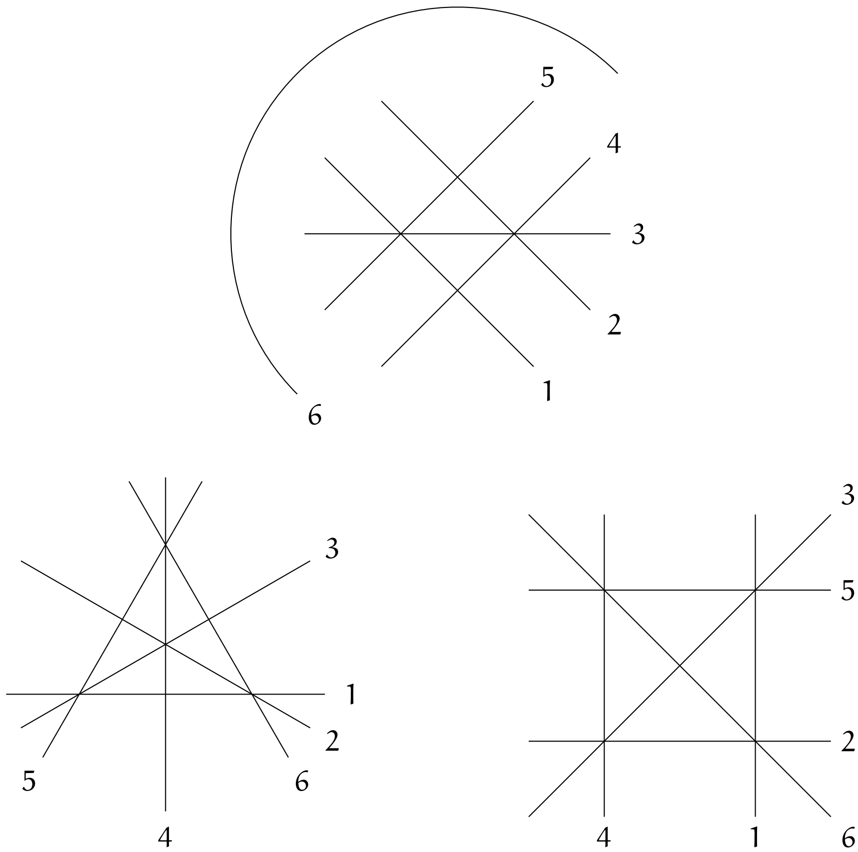
The catalogue is sorted by increasing number of lines. In the pictures, an arc around an arrangement means that the line at infinity belongs to the arrangement.

Below each arrangement there are the list of neighbourly partitions (NP), the double points graph (DPG; see Definition 2.8), and the list of the components of the characteristic variety (CV). The NP list includes partitions of subarrangements and it is organized by type of the subarrangement; however, NP corresponding to singular points are *not* reported here.

In the CV list, only equations for essential components are shown. Equations for the other components can be derived from the corresponding components in the subarrangements (see Proposition 3.30 and the subsequent discussion for translated components).

You will notice that, if an arrangement has n lines, the equations displayed here are (Laurent) polynomials in $n-1$ variables. In fact, to save computation time, all the characteristic varieties have been computed starting from the *affine* arrangement with $n-1$ lines obtained by sending the line n (i.e. the one with maximum index in the pictures) to infinity. Equations for the actual components can be computed by adding $t_1 \cdots t_n - 1$ to the equations here (see Theorem 3.14).

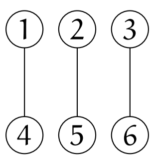
A3



Alias(es): Braid [45], $\mathcal{A}(6,1)$ [17, 28], $\mathcal{R}(6)$ [29], $\mathcal{B}_6$ [47], $\mathcal{Q}(2)$
NP:

- (essential): $\{\{1,4\},\{2,5\},\{3,6\}\}$

DPG:

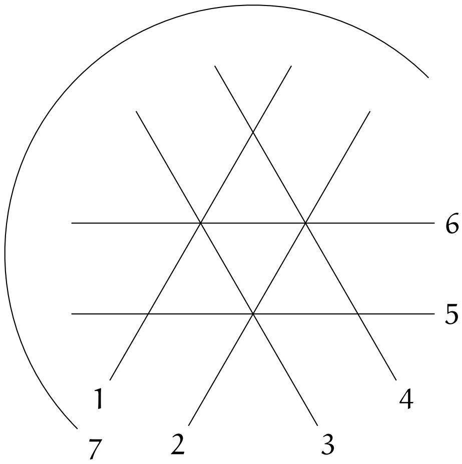


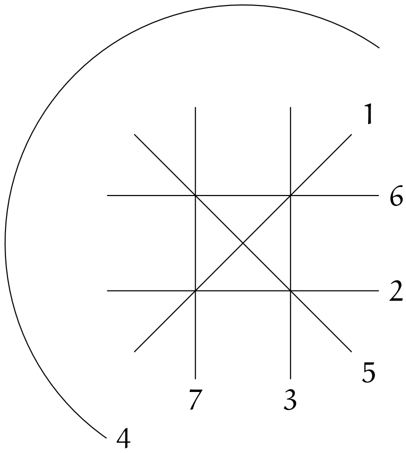
CV:

- 4 local components (= 4 triple pts.)
- 1 essential 2-dimensional component:

$$t_1 - t_4, \qquad t_2 - t_5, \qquad t_3 t_4 t_5 - 1$$

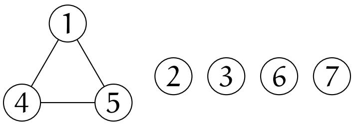
NonFano





Alias(es): $\mathcal{A}(7,1)$ [17, 28]
NP:

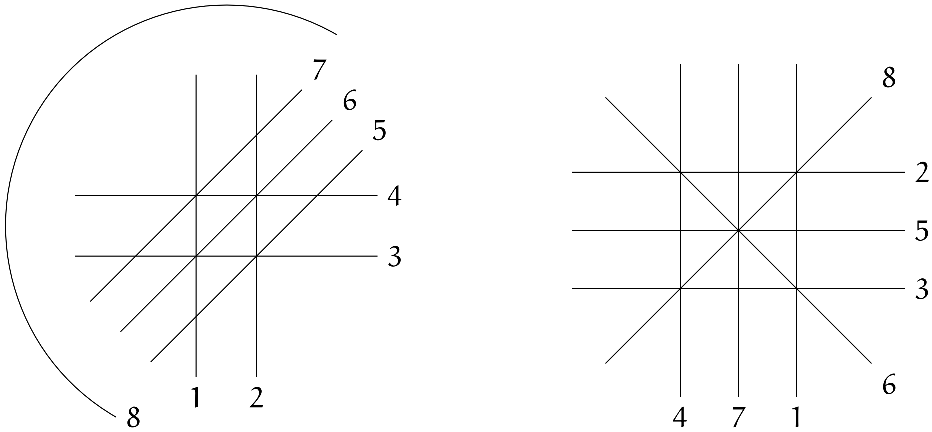
- A3: $\{\{1,4\},\{2,3\},\{6,7\}\}$, $\{\{1,5\},\{2,6\},\{3,7\}\}$, $\{\{2,7\},\{3,6\},\{4,5\}\}$
- (essential): $\{\{1,4,5\},\{2\},\{3\},\{6\},\{7\}\}$

DPG: 

CV:

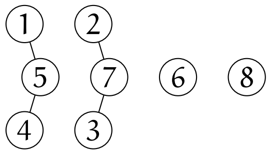
- 6 local components (= 6 triple pts.)
- 3 components of type $A3$

B3x



Alias(es): $\mathcal{A}(8,1)$ [17, 28], B3 deleted [26, 45, 48], $\mathcal{D}_2$ [7], $\mathcal{Q}(3)$, Cohen(2)
NP:

- $A3$: $\{\{1,4\},\{2,3\},\{6,8\}\}$, $\{\{1,5\},\{2,6\},\{3,8\}\}$, $\{\{1,6\},\{2,7\},\{4,8\}\}$, $\{\{1,8\},\{3,7\},\{4,6\}\}$, $\{\{2,8\},\{3,6\},\{4,5\}\}$
- NonFano: $\{\{1,4,5\},\{2\},\{3\},\{6\},\{8\}\}$, $\{\{1\},\{2,3,7\},\{4\},\{6\},\{8\}\}$

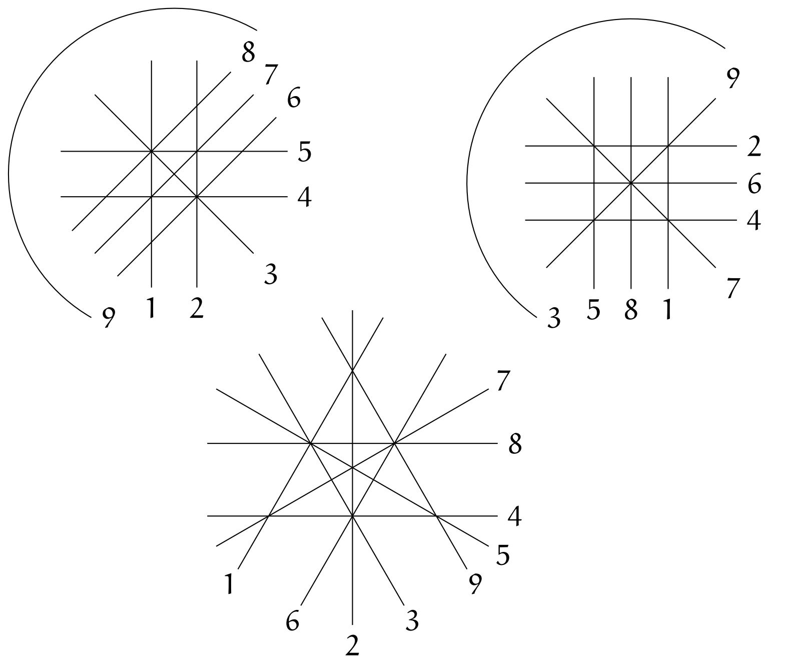
DPG: 

CV:

- 7 local components (= 6 triple pts. + 1 quadruple pt.)
- 5 components of type $A3$
- 1 essential 1-dimensional translated component:

$$t_6+1, \quad t_2-t_3, \quad t_1-t_4, \quad t_5t_7-1, \quad t_4t_7+t_3,$$
$$t_3t_5+t_4, \quad t_4^2-t_5, \quad t_3t_4+1, \quad t_3^2-t_7$$

B3

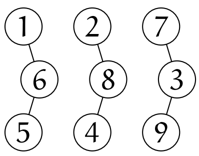


Alias(es): $\mathcal{A}(9,1)$ [17, 28]

NP:

- $A3$: $\{\{1,2\},\{3,7\},\{4,5\}\}$, $\{\{1,4\},\{2,5\},\{3,9\}\}$, $\{\{1,6\},\{2,8\},\{3,9\}\}$, $\{\{1,5\},\{2,4\},\{7,9\}\}$, $\{\{1,6\},\{2,7\},\{4,9\}\}$, $\{\{1,7\},\{2,8\},\{5,9\}\}$, $\{\{1,6\},\{3,7\},\{4,8\}\}$, $\{\{1,9\},\{4,8\},\{5,7\}\}$, $\{\{2,8\},\{3,7\},\{5,6\}\}$, $\{\{2,9\},\{4,7\},\{5,6\}\}$, $\{\{3,9\},\{4,8\},\{5,6\}\}$
- NonFano: $\{\{1\},\{2\},\{3,7,9\},\{4\},\{5\}\}$, $\{\{1,5,6\},\{2\},\{4\},\{7\},\{9\}\}$, $\{\{1\},\{2,4,8\},\{5\},\{7\},\{9\}\}$
- (essential): $\{\{1,5,6\},\{2,4,8\},\{3,7,9\}\}$

DPG:

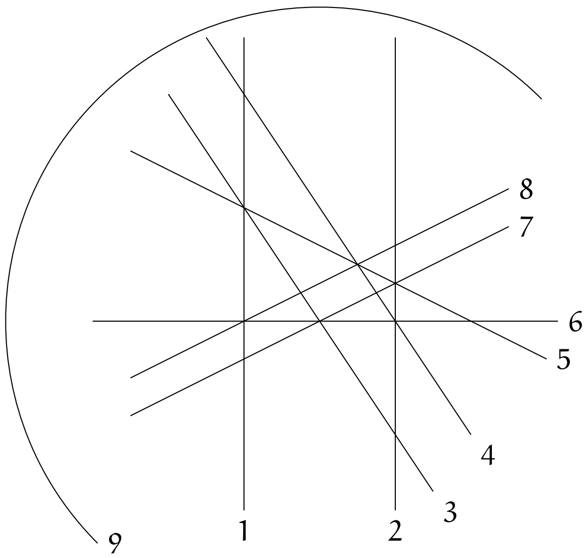


CV:

- 7 local components (= 4 triple pts. + 3 quadruple pts.)
- 11 components of type $A3$
- 1 essential 2-dimensional component:

$$t_2 - t_4, \quad t_1 - t_5, \quad t_7^2 - t_3, \quad t_5^2 - t_6, \quad t_4^2 - t_8, \quad t_6t_7t_8 - t_4t_5,$$
$$t_5t_7t_8 - t_4, \quad t_3t_6t_8 - 1, \quad t_3t_5t_8 - t_4t_7, \quad t_4t_6t_7 - t_5,$$
$$t_4t_5t_7 - 1, \quad t_3t_4t_6 - t_5t_7, \quad t_3t_4t_5 - t_7$$

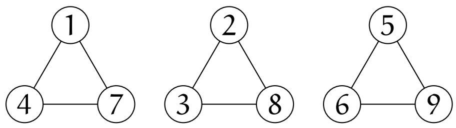
Pappus



NP:

- (essential): $\{\{1,4,7\},\{2,3,8\},\{5,6,9\}\}$

DPG:

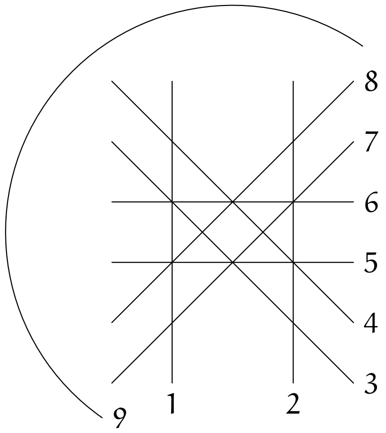


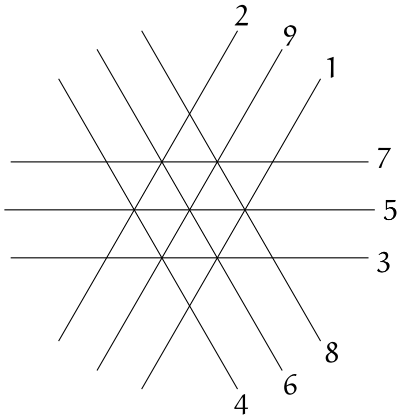
CV:

- 9 local components (= 9 triple pts.)
- 1 essential 2-dimensional component:

$$t_5 - t_6, \quad t_4 - t_7, \quad t_3 - t_8, \quad t_2 - t_8, \quad t_1 - t_7, \quad t_6 t_7 t_8 - 1$$

NonPappus

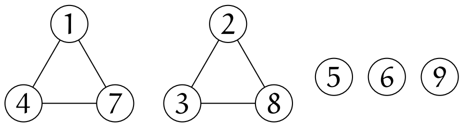




Alias(es): Pappus[▸1] [24]
Note: This arrangement is *not* the "non-Pappus" arrangement of [45]. In fact, in [45] both their "Pappus" and "non-Pappus" arrangements have the property that each line passes through exactly three triple points, but "the difference in their position is reflected in several invariants of the complement" ([45, p. 74]). For example, the characteristic variety of their "non-Pappus" arrangement has no non-local components. On the other hand, in our NonPappus arrangement there is a line (9 in the picture above) that passes through *four* triple points.
NP:

- (essential): $\{\{1,4,7\},\{2,3,8\},\{5,6,9\}\}$, $\{\{1,4,7\},\{2,3,8\},\{5\},\{6\},\{9\}\}$

DPG:



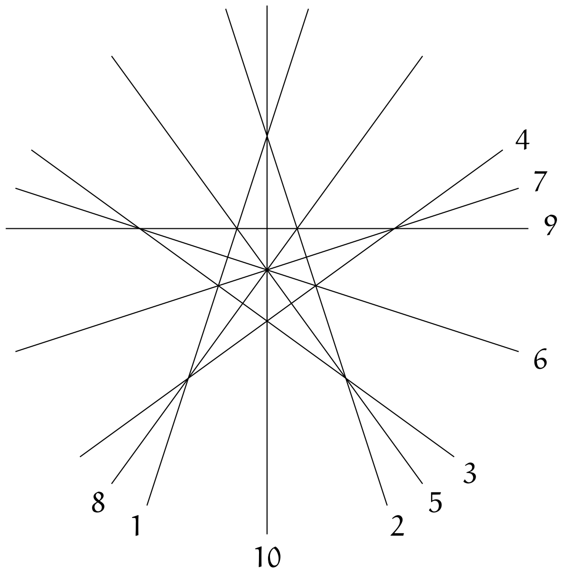
CV:

- 10 local components (= 10 triple pts.)
- 1 essential 2-dimensional component:

$$t_5 - t_6, \quad t_4 - t_7, \quad t_3 - t_8, \quad t_2 - t_8, \quad t_1 - t_7, \quad t_6t_7t_8 - 1$$

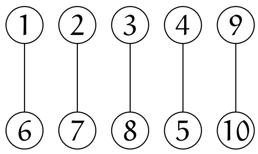
[▸1] In [24], this arrangement is called "Pappus" despite not having the property that each line passes through exactly three triple points.

$\mathcal{R}(10)$



Alias(es): $\mathcal{A}(10, 1)$ [17, 28]
NP:

- A3: $\{\{1, 5\}, \{2, 7\}, \{3, 10\}\}$, $\{\{1, 6\}, \{2, 8\}, \{4, 10\}\}$, $\{\{1, 8\}, \{2, 5\}, \{9, 10\}\}$, $\{\{1, 10\}, \{3, 8\}, \{4, 7\}\}$, $\{\{1, 6\}, \{3, 5\}, \{7, 9\}\}$, $\{\{1, 7\}, \{4, 5\}, \{8, 9\}\}$, $\{\{2, 10\}, \{3, 6\}, \{4, 5\}\}$, $\{\{2, 6\}, \{3, 8\}, \{5, 9\}\}$, $\{\{2, 7\}, \{4, 8\}, \{6, 9\}\}$, $\{\{3, 7\}, \{4, 6\}, \{9, 10\}\}$
- (essential): $\{\{1, 6\}, \{2, 7\}, \{3, 8\}, \{4, 5\}, \{9, 10\}\}$

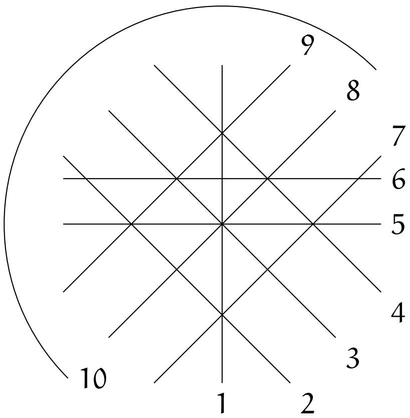
DPG: 

CV:

- 11 local components (= 10 triple pts. + 1 5-tuple pt.)
- 10 components of type A3
- 4 essential 0-dimensional translated components:

$$t_7 - t_8, \quad t_6 - t_8, \quad t_5 - t_8, \quad t_4 - t_9, \quad t_3 - t_9, \quad t_2 - t_9, \quad t_1 - t_9,$$
$$t_8 t_9 + t_9^2 + t_8 + t_9 + 1, \quad t_8^2 - t_9, \quad t_9^3 - t_8$$

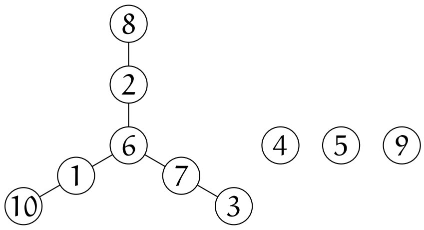
$\mathcal{A}(10, 2)$



NP:

- A3: {{1, 4}, {2, 5}, {3, 7}}, {{1, 2}, {3, 9}, {4, 5}}, {{1, 10}, {2, 8}, {3, 7}}, {{1, 5}, {2, 4}, {7, 9}}, {{1, 10}, {2, 9}, {4, 7}}, {{1, 9}, {2, 8}, {5, 7}}, {{1, 6}, {3, 4}, {8, 9}}, {{1, 10}, {3, 9}, {4, 8}}, {{1, 7}, {4, 8}, {5, 9}}, {{2, 6}, {3, 5}, {9, 10}}, {{2, 8}, {3, 9}, {5, 10}}, {{2, 7}, {4, 9}, {5, 10}}, {{3, 6}, {4, 5}, {8, 10}}, {{3, 7}, {4, 8}, {5, 10}}, {{3, 8}, {4, 9}, {6, 10}}, {{3, 10}, {5, 9}, {6, 8}}, {{4, 10}, {5, 8}, {6, 7}}
- NonFano: {{1}, {2}, {3, 7, 9}, {4}, {5}}, {{1}, {2, 4, 8}, {5}, {7}, {9}}, {{1, 5, 10}, {2}, {4}, {7}, {9}}, {{1, 6, 10}, {3}, {4}, {8}, {9}}, {{2, 6, 8}, {3}, {5}, {9}, {10}}, {{3, 6, 7}, {4}, {5}, {8}, {10}}, {{3}, {4, 5, 9}, {6}, {8}, {10}}
- B3: {{1, 5, 10}, {2, 4, 8}, {3, 7, 9}}

DPG:

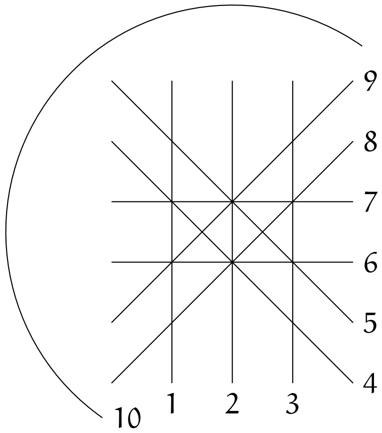


CV:

- 10 local components (= 7 triple pts. + 3 quadruple pts.)
- 17 components of type A3
- 1 component of type B3
- 3 translated components of type B3x
- 2 essential 0-dimensional translated components:

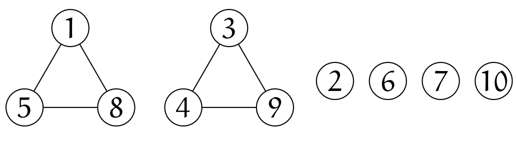
$$t_8 - t_9 - 1, \quad t_7 - t_9, \quad t_6 + 1, \quad t_5 - t_9, \quad t_4 - t_9,$$
$$t_3 - t_9 - 1, \quad t_2 - t_9, \quad t_1 - t_9, \quad t_9^2 + t_9 + 1$$

$\mathcal{A}(10, 3)$



NP:

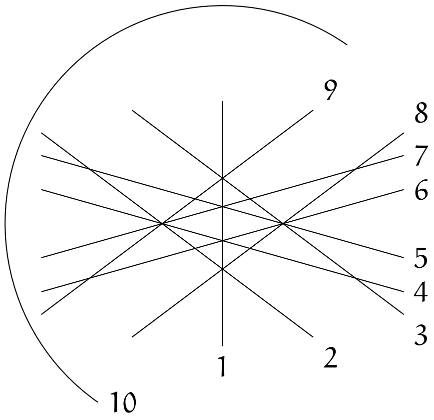
- A3: $\{\{1,8\},\{2,7\},\{3,4\}\}$, $\{\{1,5\},\{2,6\},\{3,9\}\}$, $\{\{1,5\},\{2,4\},\{7,10\}\}$, $\{\{1,2\},\{4,9\},\{6,7\}\}$, $\{\{1,6\},\{2,7\},\{4,10\}\}$, $\{\{1,7\},\{2,6\},\{9,10\}\}$, $\{\{1,8\},\{2,9\},\{6,10\}\}$, $\{\{2,5\},\{3,4\},\{6,10\}\}$, $\{\{2,3\},\{5,8\},\{6,7\}\}$, $\{\{2,6\},\{3,7\},\{5,10\}\}$, $\{\{2,7\},\{3,6\},\{8,10\}\}$, $\{\{2,8\},\{3,9\},\{7,10\}\}$, $\{\{2,10\},\{4,7\},\{5,6\}\}$, $\{\{2,10\},\{4,9\},\{5,8\}\}$, $\{\{2,10\},\{6,9\},\{7,8\}\}$
- NonFano: $\{\{1,5,6\},\{2\},\{4\},\{7\},\{10\}\}$, $\{\{1\},\{2\},\{4,9,10\},\{6\},\{7\}\}$, $\{\{1,7,8\},\{2\},\{6\},\{9\},\{10\}\}$, $\{\{2\},\{3,4,7\},\{5\},\{6\},\{10\}\}$, $\{\{2\},\{3\},\{5,8,10\},\{6\},\{7\}\}$, $\{\{2\},\{3,6,9\},\{7\},\{8\},\{10\}\}$
- NonPappus: $\{\{1,5,8\},\{3,4,9\},\{6,7,10\}\}$, $\{\{1,5,8\},\{3,4,9\},\{6\},\{7\},\{10\}\}$
- (essential): $\{\{1,5,8\},\{2,6,7,10\},\{3,4,9\}\}$, $\{\{1,5,8\},\{2,3,4,9\},\{6,7,10\}\}$, $\{\{1,2,5,8\},\{3,4,9\},\{6,7,10\}\}$, $\{\{1,5,8\},\{2,3,4,9\},\{6\},\{7\},\{10\}\}$, $\{\{1,2,5,8\},\{3,4,9\},\{6\},\{7\},\{10\}\}$, $\{\{1,5,8\},\{2\},\{3,4,9\},\{6,7,10\}\}$, $\{\{1,5,8\},\{2,10\},\{3,4,9\},\{6\},\{7\}\}$, $\{\{1,5,8\},\{2,7\},\{3,4,9\},\{6\},\{10\}\}$, $\{\{1,5,8\},\{2,6\},\{3,4,9\},\{7\},\{10\}\}$, $\{\{1,5,8\},\{2\},\{3,4,9\},\{6\},\{7\},\{10\}\}$

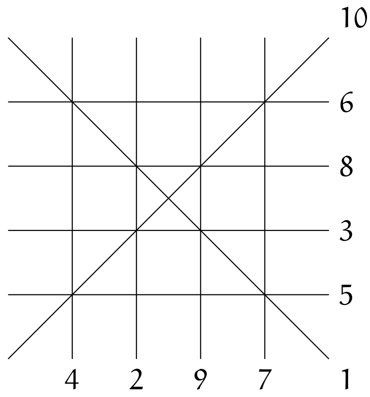
DPG: 

CV:

- 10 local components (= 7 triple pts. + 3 quadruple pts.)
- 15 components of type A3
- 1 component of type NonPappus
- 6 translated components of type B3x

$\mathcal{Q}(4)$

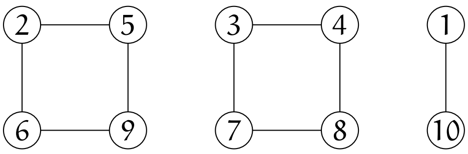




NP:

- $A3$: $\{\{1,10\},\{2,9\},\{3,8\}\}$, $\{\{1,10\},\{4,7\},\{5,6\}\}$
- (essential): $\{\{1,10\},\{2,5,6,9\},\{3,4,7,8\}\}$

DPG:

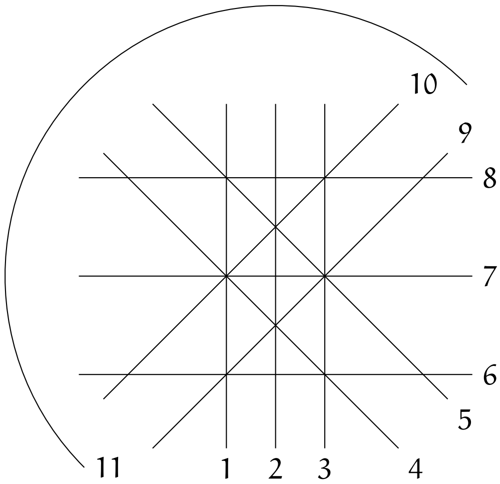


CV:

- 10 local components (= 8 triple pts. + 2 quadruple pts.)
- 2 components of type $A3$
- 1 essential translated 1-dimensional component:

$$t_7 - t_8,\quad t_6 - t_9,\quad t_5 - t_9,\quad t_4 - t_8,$$
$$t_3 - t_8,\quad t_2 - t_9,\quad t_1 + 1,\quad t_8 t_9 + 1$$

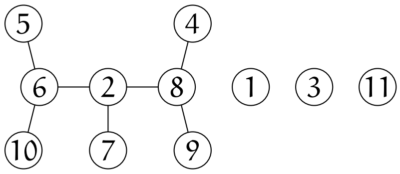
$\mathcal{A}(11,1)$



NP:

- A3: $\{\{1,4\},\{2,6\},\{3,9\}\}$, $\{\{1,9\},\{2,7\},\{3,4\}\}$, $\{\{1,5\},\{2,7\},\{3,10\}\}$, $\{\{1,10\},\{2,8\},\{3,5\}\}$, $\{\{1,5\},\{2,4\},\{10,11\}\}$, $\{\{1,9\},\{2,10\},\{4,11\}\}$, $\{\{1,5\},\{3,4\},\{7,11\}\}$, $\{\{1,3\},\{4,9\},\{6,7\}\}$, $\{\{1,6\},\{3,7\},\{4,11\}\}$, $\{\{1,3\},\{5,10\},\{7,8\}\}$, $\{\{1,7\},\{3,8\},\{5,11\}\}$, $\{\{1,7\},\{3,6\},\{9,11\}\}$, $\{\{1,8\},\{3,7\},\{10,11\}\}$, $\{\{1,9\},\{3,10\},\{7,11\}\}$, $\{\{1,11\},\{4,8\},\{5,7\}\}$, $\{\{1,7\},\{5,6\},\{8,9\}\}$, $\{\{1,11\},\{6,10\},\{7,9\}\}$, $\{\{2,5\},\{3,4\},\{9,11\}\}$, $\{\{2,9\},\{3,10\},\{5,11\}\}$, $\{\{2,7\},\{4,5\},\{9,10\}\}$, $\{\{2,11\},\{4,10\},\{5,9\}\}$, $\{\{3,11\},\{4,7\},\{5,6\}\}$, $\{\{3,7\},\{4,8\},\{6,10\}\}$, $\{\{3,11\},\{7,10\},\{8,9\}\}$, $\{\{4,9\},\{5,10\},\{7,11\}\}$
- NonFano: $\{\{1\},\{2,6,7\},\{3\},\{4\},\{9\}\}$, $\{\{1\},\{2,7,8\},\{3\},\{5\},\{10\}\}$, $\{\{1,5,9\},\{2\},\{4\},\{10\},\{11\}\}$, $\{\{1,5,6\},\{3\},\{4\},\{7\},\{11\}\}$, $\{\{1\},\{3,4,8\},\{5\},\{7\},\{11\}\}$, $\{\{1\},\{3\},\{4,9,11\},\{6\},\{7\}\}$, $\{\{1\},\{3\},\{5,10,11\},\{7\},\{8\}\}$, $\{\{1\},\{3,6,10\},\{7\},\{9\},\{11\}\}$, $\{\{1,8,9\},\{3\},\{7\},\{10\},\{11\}\}$, $\{\{2\},\{3,4,10\},\{5\},\{9\},\{11\}\}$, $\{\{2,7,11\},\{4\},\{5\},\{9\},\{10\}\}$
- B3: $\{\{1,5,9\},\{2,7,11\},\{3,4,10\}\}$
- NonPappus: $\{\{1,3,11\},\{4,8,9\},\{5,6,10\}\}$, $\{\{1\},\{3\},\{4,8,9\},\{5,6,10\},\{11\}\}$
- $\mathcal{A}(10,3)$: $\{\{1,3,11\},\{4,8,9\},\{5,6,7,10\}\}$, $\{\{1,3,11\},\{4,7,8,9\},\{5,6,10\}\}$, $\{\{1,3,7,11\},\{4,8,9\},\{5,6,10\}\}$, $\{\{1\},\{3\},\{4,8,9\},\{5,6,7,10\},\{11\}\}$, $\{\{1\},\{3\},\{4,7,8,9\},\{5,6,10\},\{11\}\}$, $\{\{1,3,11\},\{4,8,9\},\{5,6,10\},\{7\}\}$, $\{\{1\},\{3\},\{4,8,9\},\{5,6,10\},\{7,11\}\}$, $\{\{1\},\{3,7\},\{4,8,9\},\{5,6,10\},\{11\}\}$, $\{\{1,7\},\{3\},\{4,8,9\},\{5,6,10\},\{11\}\}$, $\{\{1\},\{3\},\{4,8,9\},\{5,6,10\},\{7\},\{11\}\}$

DPG:

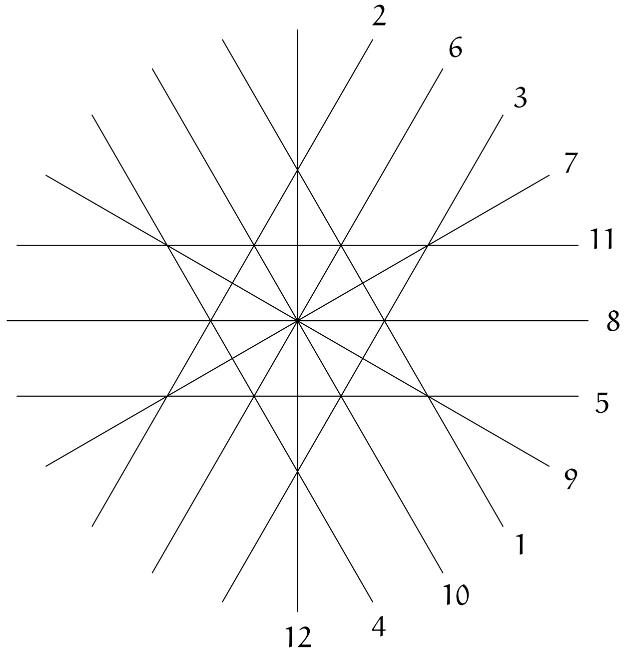


CV:

- 12 local components (= 8 triple pts. + 4 quadruple pts.)
- 25 components of type A3
- 1 component of type B3
- 1 component of type NonPappus
- 8 translated components of type B3x
- 4 translated components of type $\mathcal{A}(10, 2)$
- 2 essential 0-dimensional translated components:

$$t_9 - t_{10}, \quad t_8 + 1, \quad t_7 - t_{10} + 1, \quad t_6 + 1, \quad t_5 - t_{10}, \quad t_4 - t_{10},$$
$$t_3 - t_{10} + 1, \quad t_2 + t_{10}, \quad t_1 - t_{10} + 1, \quad t_{10}^2 - t_{10} + 1$$

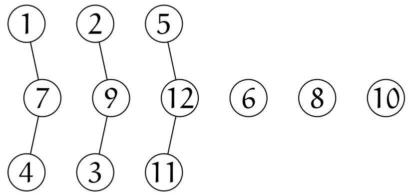
$\mathcal{R}(12)$



NP:

- A3: {{1, 4}, {2, 3}, {8, 12}}, {{1, 6}, {2, 8}, {3, 12}}, {{1, 8}, {2, 10}, {4, 12}}, {{1, 7}, {2, 9}, {5, 12}}, {{1, 10}, {2, 6}, {11, 12}}, {{1, 12}, {3, 10}, {4, 8}}, {{1, 10}, {3, 9}, {5, 8}}, {{1, 7}, {3, 6}, {8, 11}}, {{1, 6}, {4, 9}, {5, 10}}, {{1, 4}, {5, 11}, {6, 9}}, {{1, 9}, {4, 6}, {10, 11}}, {{1, 8}, {5, 6}, {9, 11}}, {{2, 12}, {3, 8}, {4, 6}}, {{2, 10}, {3, 7}, {5, 6}}, {{2, 3}, {5, 11}, {7, 10}}, {{2, 7}, {3, 10}, {6, 11}}, {{2, 6}, {4, 7}, {5, 8}}, {{2, 9}, {4, 10}, {8, 11}}, {{2, 8}, {5, 10}, {7, 11}}, {{3, 6}, {4, 10}, {5, 12}}, {{3, 9}, {4, 7}, {11, 12}}, {{3, 8}, {5, 7}, {10, 11}}, {{4, 8}, {5, 9}, {6, 11}}
- NonFano: {{1, 4, 6}, {2}, {3}, {8}, {12}}, {{1}, {2, 3, 10}, {4}, {8}, {12}}, {{1, 4, 8}, {5}, {6}, {9}, {11}}, {{1}, {4}, {5, 10, 11}, {6}, {9}}, {{2}, {3}, {5, 6, 11}, {7}, {10}}, {{2, 3, 8}, {5}, {7}, {10}, {11}}
- NonPappus: {{1, 2, 5}, {3, 4, 11}, {6, 8, 10}}, {{1, 2, 5}, {3, 4, 11}, {6}, {8}, {10}}
- (essential): {{1, 4, 7, 10}, {2, 3, 6, 9}, {5, 8, 11, 12}}

DPG:



CV:

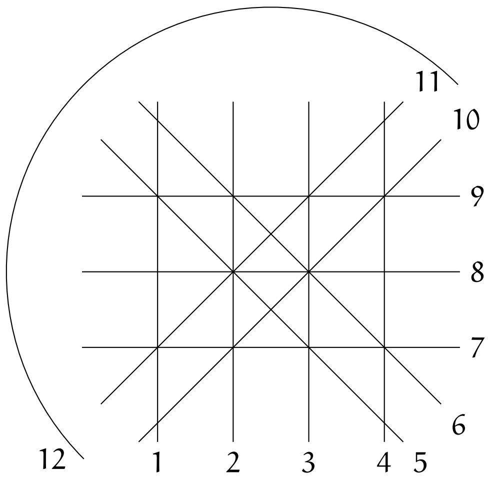
- 16 local components (= 15 triple pts. + 1 6-tuple pt.)
- 23 components of type A3
- 1 component of type NonPappus
- 3 translated components of type B3x
- 1 essential 2-dimensional component:

$$t_8 - t_{11}, \quad t_7 - t_{10}, \quad t_6 - t_9, \quad t_5 - t_{11}, \quad t_4 - t_{10},$$
$$t_3 - t_9, \quad t_2 - t_9, \quad t_1 - t_{10}, \quad t_9 t_{10} t_{11} - 1$$

- 2 essential 0-dimensional translated components:

$$t_{10} + t_{11}, \quad t_9 + t_{11}, \quad t_8 + t_{11}, \quad t_7 + t_{11}, \quad t_6 + t_{11}, \quad t_5 - t_{11}, \quad t_4 - t_{11},$$
$$t_3 - t_{11}, \quad t_2 - t_{11}, \quad t_1 - t_{11}, \quad t_{11}^2 - t_{11} + 1$$

$\mathcal{A}(12, 2)$



NP:

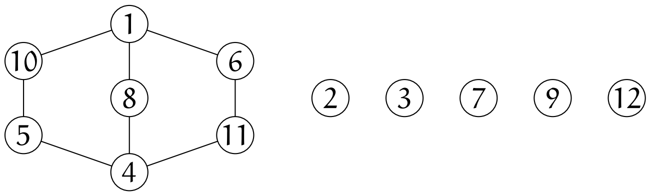
- A3: {{1, 5}, {2, 7}, {3, 11}}, {{1, 11}, {2, 9}, {3, 5}}, {{1, 6}, {2, 5}, {9, 12}}, {{1, 8}, {2, 9}, {5, 12}}, {{1, 8}, {2, 7}, {11, 12}}, {{1, 10}, {2, 11}, {7, 12}}, {{1, 3}, {5, 11}, {7, 9}}, {{1, 7}, {3, 9}, {5, 12}}, {{1, 9}, {3, 7}, {11, 12}}, {{1, 8}, {5, 7}, {9, 11}}, {{2, 6}, {3, 7}, {4, 10}}, {{2, 10}, {3, 9}, {4, 6}}, {{2, 6}, {3, 5}, {8, 12}}, {{2, 3}, {5, 10}, {7, 8}}, {{2, 7}, {3, 8}, {5, 12}}, {{2, 3}, {6, 11}, {8, 9}}, {{2, 8}, {3, 9}, {6, 12}}, {{2, 8}, {3, 7}, {10, 12}}, {{2, 9}, {3, 8}, {11, 12}}, {{2, 10}, {3, 11}, {8, 12}}, {{2, 4}, {6, 10}, {7, 9}}, {{2, 7}, {4, 9}, {6, 12}}, {{2, 9}, {4, 7}, {10, 12}}, {{2, 12}, {5, 9}, {6, 8}}, {{2, 8}, {6, 7}, {9, 10}}, {{2, 12}, {7, 11}, {8, 10}}, {{3, 6}, {4, 5}, {7, 12}}, {{3, 7}, {4, 8}, {6, 12}}, {{3, 9}, {4, 8}, {10, 12}}, {{3, 10}, {4, 11}, {9, 12}}, {{3, 12}, {5, 8}, {6, 7}}, {{3, 8}, {5, 9}, {7, 11}}, {{3, 12}, {8, 11}, {9, 10}}, {{4, 8}, {6, 9}, {7, 10}}, {{5, 10}, {6, 11}, {8, 12}}

- NonFano: {{1}, {2, 7, 9}, {3}, {5}, {11}}, {{1, 6, 8}, {2}, {5}, {9}, {12}}, {{1, 8, 10}, {2}, {7}, {11}, {12}}, {{1, 3, 8}, {5}, {7}, {9}, {11}}, {{1}, {3}, {5, 11, 12}, {7}, {9}}, {{2}, {3, 7, 9}, {4}, {6}, {10}}, {{2, 6, 7}, {3}, {5}, {8}, {12}}, {{2}, {3, 5, 9}, {6}, {8}, {12}}, {{2}, {3}, {5, 10, 12}, {7}, {8}}, {{2}, {3}, {6, 11, 12}, {8}, {9}}, {{2}, {3, 7, 11}, {8}, {10}, {12}}, {{2, 9, 10}, {3}, {8}, {11}, {12}}, {{2, 4, 8}, {6}, {7}, {9}, {10}}, {{2}, {4}, {6, 10, 12}, {7}, {9}}, {{3}, {4, 5, 8}, {6}, {7}, {12}}, {{3}, {4, 8, 11}, {9}, {10}, {12}}

- B3: {{1, 3, 8}, {2, 7, 9}, {5, 11, 12}}, {{2, 4, 8}, {3, 7, 9}, {6, 10, 12}}

- NonPappus: $\{\{2,3,12\},\{5,9,10\},\{6,7,11\}\}$, $\{\{2\},\{3\},\{5,9,10\},\{6,7,11\},\{12\}\}$
- $\mathcal{A}(10,3)$: $\{\{2,3,12\},\{5,9,10\},\{6,7,8,11\}\}$, $\{\{2,3,12\},\{5,8,9,10\},\{6,7,11\}\}$, $\{\{2,3,8,12\},\{5,9,10\},\{6,7,11\}\}$, $\{\{2\},\{3\},\{5,9,10\},\{6,7,8,11\},\{12\}\}$, $\{\{2\},\{3\},\{5,8,9,10\},\{6,7,11\},\{12\}\}$, $\{\{2,3,12\},\{5,9,10\},\{6,7,11\},\{8\}\}$, $\{\{2\},\{3\},\{5,9,10\},\{6,7,11\},\{8,12\}\}$, $\{\{2\},\{3,8\},\{5,9,10\},\{6,7,11\},\{12\}\}$, $\{\{2,8\},\{3\},\{5,9,10\},\{6,7,11\},\{12\}\}$, $\{\{2\},\{3\},\{5,9,10\},\{6,7,11\},\{8\},\{12\}\}$

DPG:

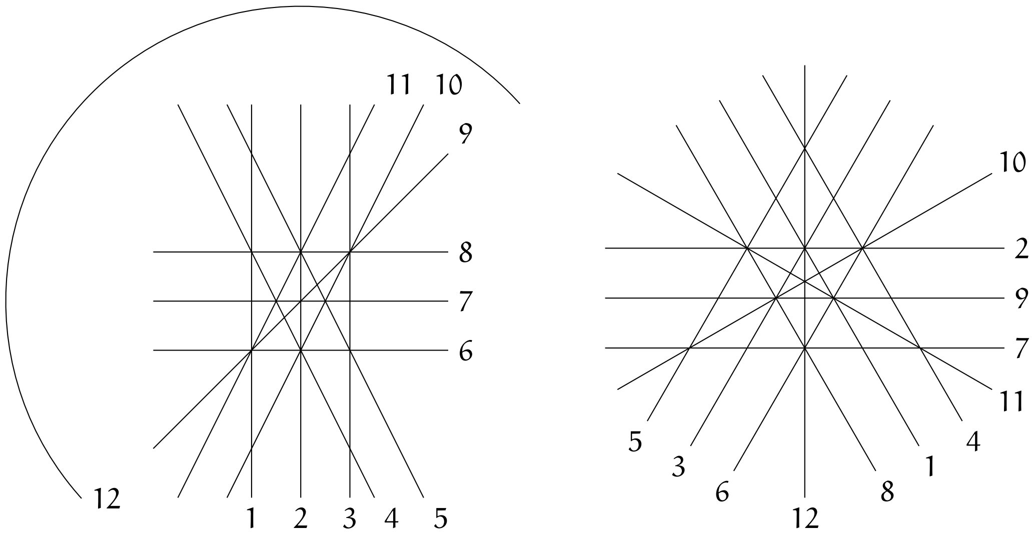


CV:

- 14 local components (= 10 triple pts. + 3 quadruple pts. + 1 5-tuple pt.)
- 35 components of type A3
- 2 components of type B3
- 1 component of type NonPappus
- 10 translated components of type B3x
- 8 translated components of type $\mathcal{A}(10,2)$
- 4 translated components of type $\mathcal{A}(11,1)$
- 2 essential 0-dimensional translated components:

$$t_{10}-t_{11},\quad t_9+1,\quad t_8-t_{11}+1,\quad t_7+1,\quad t_6-t_{11},\quad t_5-t_{11},$$
$$t_4-t_{11}+1,\quad t_3-t_{11}+1,\quad t_2-t_{11}+1,\quad t_1-t_{11}+1,\quad t_{11}^2-t_{11}+1$$

$\mathcal{A}(12, 3)$



NP:

- A3: {{1, 10}, {2, 8}, {3, 4}}, {{1, 5}, {2, 6}, {3, 11}}, {{1, 10}, {2, 9}, {3, 6}}, {{1, 8}, {2, 9}, {3, 11}}, {{1, 5}, {2, 4}, {8, 12}}, {{1, 2}, {4, 11}, {6, 8}}, {{1, 6}, {2, 8}, {4, 12}}, {{1, 7}, {2, 6}, {9, 12}}, {{1, 8}, {2, 6}, {11, 12}}, {{1, 10}, {2, 11}, {6, 12}}, {{1, 8}, {3, 6}, {9, 12}}, {{1, 10}, {3, 11}, {9, 12}}, {{1, 7}, {4, 6}, {8, 11}}, {{1, 10}, {4, 9}, {6, 8}}, {{2, 5}, {3, 4}, {6, 12}}, {{2, 3}, {5, 10}, {6, 8}}, {{2, 6}, {3, 8}, {5, 12}}, {{2, 8}, {3, 6}, {10, 12}}, {{2, 8}, {3, 7}, {9, 12}}, {{2, 10}, {3, 11}, {8, 12}}, {{2, 12}, {4, 8}, {5, 6}}, {{2, 7}, {4, 5}, {10, 11}}, {{2, 12}, {4, 11}, {5, 10}}, {{2, 7}, {4, 8}, {6, 11}}, {{2, 11}, {4, 9}, {6, 7}}, {{2, 7}, {5, 6}, {8, 10}}, {{2, 10}, {5, 9}, {7, 8}}, {{2, 8}, {6, 9}, {7, 10}}, {{2, 6}, {7, 11}, {8, 9}}, {{2, 9}, {6, 8}, {10, 11}}, {{2, 12}, {6, 11}, {8, 10}}, {{3, 7}, {5, 8}, {6, 10}}, {{3, 11}, {5, 9}, {6, 8}}, {{4, 7}, {5, 6}, {10, 12}}, {{4, 8}, {5, 7}, {11, 12}}, {{4, 10}, {5, 11}, {7, 12}}, {{4, 12}, {6, 11}, {7, 10}}, {{5, 12}, {7, 11}, {8, 10}}, {{6, 10}, {8, 11}, {9, 12}}

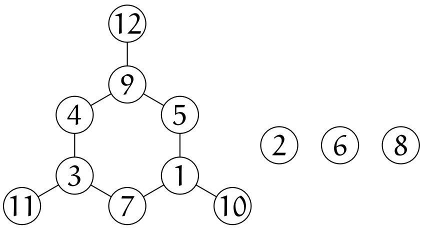
- NonFano: {{1, 5, 6}, {2}, {4}, {8}, {12}}, {{1, 2, 7}, {4}, {6}, {8}, {11}}, {{1}, {2}, {4, 11, 12}, {6}, {8}}, {{1, 8, 10}, {2}, {6}, {11}, {12}}, {{2}, {3, 4, 8}, {5}, {6}, {12}}, {{2, 3, 7}, {5}, {6}, {8}, {10}}, {{2}, {3}, {5, 10, 12}, {6}, {8}}, {{2}, {3, 6, 11}, {8}, {10}, {12}}, {{2, 7, 12}, {4}, {5}, {10}, {11}}, {{2}, {4, 8, 9}, {6}, {7}, {11}}, {{2}, {5, 6, 9}, {7}, {8}, {10}}, {{2}, {6}, {7, 10, 11}, {8}, {9}}, {{2, 9, 12}, {6}, {8}, {10}, {11}}, {{4}, {5, 6, 11}, {7}, {10}, {12}}, {{4, 8, 10}, {5}, {7}, {11}, {12}}

- B3; {{1, 8, 10}, {2, 9, 12}, {3, 6, 11}}, {{2, 7, 12}, {4, 8, 10}, {5, 6, 11}}

- NonPappus: {{1, 7, 10}, {2, 8, 11}, {4, 9, 12}}, {{1, 7, 10}, {2}, {4, 9, 12}, {8}, {11}}, {{1, 5, 10}, {3, 4, 11}, {6, 8, 12}}, {{1, 5, 10}, {3, 4, 11}, {6}, {8}, {12}}, {{2, 6, 10}, {3, 7, 11}, {5, 9, 12}}, {{2}, {3, 7, 11}, {5, 9, 12}, {6}, {10}}

- $\mathcal{A}(10, 3)$: {{1, 5, 10}, {2, 6, 8, 12}, {3, 4, 11}}, {{1, 5, 10}, {2, 3, 4, 11}, {6, 8, 12}}, {{1, 2, 5, 10}, {3, 4, 11}, {6, 8, 12}}, {{1, 5, 10}, {2}, {3, 4, 11}, {6, 8, 12}}, {{1, 5, 10}, {2, 3, 4, 11}, {6}, {8}, {12}}, {{1, 5, 10}, {2, 12}, {3, 4, 11}, {6}, {8}}, {{1, 5, 10}, {2, 8}, {3, 4, 11}, {6}, {12}}, {{1, 5, 10}, {2, 6}, {3, 4, 11}, {8}, {12}}, {{1, 2, 5, 10}, {3, 4, 11}, {6}, {8}, {12}}, {{1, 5, 10}, {2}, {3, 4, 11}, {6}, {8}, {12}}, {{1, 7, 10}, {2, 6, 8, 11}, {4, 9, 12}}, {{1, 7, 10}, {2, 8, 11}, {4, 6, 9, 12}}, {{1, 6, 7, 10}, {2, 8, 11}, {4, 9, 12}}, {{1, 7, 10}, {2, 8, 11}, {4, 9, 12}, {6}}, {{1, 7, 10}, {2, 6}, {4, 9, 12}, {8}, {11}}, {{1, 7, 10}, {2}, {4, 6, 9, 12}, {8}, {11}}, {{1, 7, 10}, {2}, {4, 9, 12}, {6, 8}, {11}}, {{1, 7, 10}, {2}, {4, 9, 12}, {6, 11}, {8}}, {{1, 6, 7, 10}, {2}, {4, 9, 12}, {8}, {11}}, {{1, 7, 10}, {2}, {4, 9, 12}, {6}, {8}, {11}}, {{2, 6, 8, 10}, {3, 7, 11}, {5, 9, 12}}, {{2, 6, 10}, {3, 7, 8, 11}, {5, 9, 12}}, {{2, 6, 10}, {3, 7, 11}, {5, 8, 9, 12}}, {{2, 6, 10}, {3, 7, 11}, {5, 9, 12}, {8}}, {{2, 8}, {3, 7, 11}, {5, 9, 12}, {6}, {10}}, {{2}, {3, 7, 11}, {5, 9, 12}, {6, 8}, {10}}, {{2}, {3, 7, 11}, {5, 9, 12}, {6}, {8, 10}}, {{2}, {3, 7, 8, 11}, {5, 9, 12}, {6}, {10}}, {{2}, {3, 7, 11}, {5, 8, 9, 12}, {6}, {10}}, {{2}, {3, 7, 11}, {5, 9, 12}, {6}, {8}, {10}}

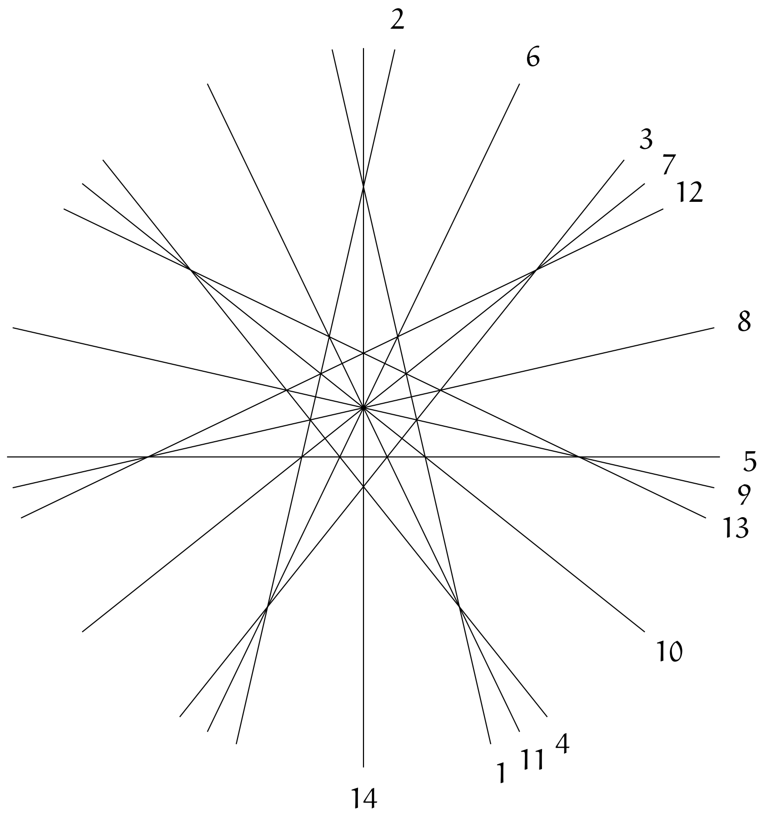
DPG:



CV:

- 13 local components (= 7 triple pts. + 6 quadruple pts.)
- 39 components of type A3
- 2 components of type B3
- 3 components of type NonPappus
- 12 translated components of type B3x
- 6 translated components of type $\mathcal{A}(10, 2)$
- 6 translated components of type $\mathcal{A}(11, 1)$

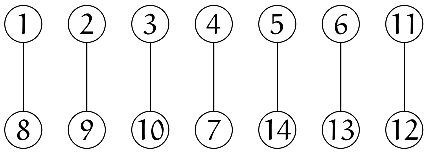
$\mathcal{R}(14)$



Alias(es): $\mathcal{A}(14,1)$ [17, 28]
NP:

- A3: $\{\{1,6\},\{2,9\},\{3,14\}\}$, $\{\{1,8\},\{2,11\},\{4,14\}\}$, $\{\{1,7\},\{2,10\},\{5,14\}\}$, $\{\{1,10\},\{2,6\},\{12,14\}\}$, $\{\{1,11\},\{2,7\},\{13,14\}\}$, $\{\{1,14\},\{3,11\},\{4,9\}\}$, $\{\{1,11\},\{3,10\},\{5,9\}\}$, $\{\{1,7\},\{3,6\},\{9,12\}\}$, $\{\{1,8\},\{3,7\},\{9,13\}\}$, $\{\{1,6\},\{4,10\},\{5,11\}\}$, $\{\{1,9\},\{4,6\},\{11,12\}\}$, $\{\{1,10\},\{4,7\},\{11,13\}\}$, $\{\{1,8\},\{5,6\},\{10,12\}\}$, $\{\{1,9\},\{5,7\},\{10,13\}\}$, $\{\{1,14\},\{6,13\},\{7,12\}\}$, $\{\{2,14\},\{3,8\},\{4,6\}\}$, $\{\{2,11\},\{3,7\},\{5,6\}\}$, $\{\{2,7\},\{3,10\},\{6,12\}\}$, $\{\{2,8\},\{3,11\},\{6,13\}\}$, $\{\{2,6\},\{4,7\},\{5,8\}\}$, $\{\{2,9\},\{4,10\},\{8,12\}\}$, $\{\{2,10\},\{4,11\},\{8,13\}\}$, $\{\{2,8\},\{5,10\},\{7,12\}\}$, $\{\{2,9\},\{5,11\},\{7,13\}\}$, $\{\{2,14\},\{10,13\},\{11,12\}\}$, $\{\{3,6\},\{4,11\},\{5,14\}\}$, $\{\{3,9\},\{4,7\},\{12,14\}\}$, $\{\{3,10\},\{4,8\},\{13,14\}\}$, $\{\{3,8\},\{5,7\},\{11,12\}\}$, $\{\{3,9\},\{5,8\},\{11,13\}\}$, $\{\{3,14\},\{7,13\},\{8,12\}\}$, $\{\{4,8\},\{5,9\},\{6,12\}\}$, $\{\{4,9\},\{5,10\},\{6,13\}\}$, $\{\{4,14\},\{9,13\},\{10,12\}\}$, $\{\{5,14\},\{8,13\},\{9,12\}\}$
- (essential): $\{\{1,8\},\{2,9\},\{3,10\},\{4,7\},\{5,14\},\{6,13\},\{11,12\}\}$

DPG:



CV:

- 22 local components (= 21 triple pts. + 1 7-tuple pt.)
- 35 components of type $A3$
- 6 essential 0-dimensional translated components:

$$t_{12}-t_{13},\quad t_{10}-t_{11},\quad t_9-t_{11},\quad t_8-t_{11},\quad t_7-t_{11},\quad t_6-t_{11},$$
$$t_5-t_{13},\quad t_4-t_{13},\quad t_3-t_{13},\quad t_2-t_{13},\quad t_1-t_{13},\quad t_{13}^3-t_{11}^2,$$
$$t_{11}t_{13}^2-1,\quad t_{11}^2t_{13}+t_{11}^2+t_{11}t_{13}+t_{13}^2+t_{11}+t_{13}+1,\quad t_{11}^3-t_{13}$$

Chapter 6

Toric Arrangements

In this chapter we depart a little from hyperplane arrangements and catch a glimpse of the theory of toric arrangements. We have encountered the algebraic torus $(\mathbb{C}^*)^n$ in Chapter 3, as it is the object where the characteristic variety of a hyperplane arrangement lives. Now we change our point of view: we consider the torus as the ambient space, in which the analogues of hyperplane arrangements are the so-called *toric arrangements*.

Also in this case we are interested in studying the complement $\mathcal{M}(\mathcal{A})$ of a toric arrangement $\mathcal{A}$. In particular, one of the research fields of this theory deals with the construction of *wonderful models* for $\mathcal{M}(\mathcal{A})$, that are smooth algebraic varieties in which $\mathcal{M}(\mathcal{A})$ embeds "nicely" (see Definition 6.13). De Concini and Gaiffi [12] recently outlined a method to obtain a projective wonderful model that requires the computation of a toric variety $X_{\mathcal{A}}$. The crucial point in this construction is provided by an algorithm that subdivides a given fan in a suitable way. Here we provide two algorithms that do so: the first one is an implementation of the algorithm described in [12], written in the `SageMath` language, and the second is an algorithm specific for the 2-dimensional case that seems to be more natural.

6.1 Toric Varieties and Toric Arrangements

We recall here some standard theory about toric varieties, mainly following [16]. This section is not intended to be a complete review of this vast theory: we just want to report here some definitions, in order to keep this work as self-contained as possible.

Definition 6.1. A *(complex algebraic) torus* is an affine variety $\mathcal{T}$ isomorphic to $(\mathbb{C}^*)^n$; it inherits the group structure through this isomorphism.

Definition 6.2. A *character* of a torus $\mathcal{T}$ is a group homomorphism $\chi\colon \mathcal{T} \to \mathbb{C}^*$ that is a morphism of algebraic varieties; the set of all characters forms a group $X^*(\mathcal{T})$ under point-wise multiplication. This group is a lattice of rank n.

Definition 6.3. A *one-parameter subgroup* of a torus $\mathcal{T}$ is a group homomorphism $\lambda\colon \mathbb{C}^* \to \mathcal{T}$ that is a morphism of algebraic varieties; the set of all one-parameter subgroups forms a group $X_*(\mathcal{T})$ under point-wise multiplication. This group is a lattice of rank n.

There is a natural $\mathbb{Z}$-bilinear pairing

$$\langle \cdot, \cdot \rangle\colon X^*(\mathcal{T}) \times X_*(\mathcal{T}) \to \mathbb{Z}$$

defined in the following way: for $\chi \in X^*(\mathcal{T})$ and $\lambda \in X_*(\mathcal{T})$, the composition $\chi \circ \lambda\colon \mathbb{C}^* \to \mathbb{C}^*$ is a character of the 1-dimensional torus $\mathbb{C}^*$, therefore it is of the form $t \mapsto t^k$ for some $k \in \mathbb{Z}$. Define $\langle \chi, \lambda \rangle := k$. This pairing allows us to identify

$$X_*(\mathcal{T}) \simeq \mathcal{H}om_{\mathbb{Z}}(X^*(\mathcal{T}), \mathbb{Z}) \quad \text{and} \quad X^*(\mathcal{T}) \simeq \mathcal{H}om_{\mathbb{Z}}(X_*(\mathcal{T}), \mathbb{Z}).$$

Moreover we define the two vector spaces

$$V := \mathcal{H}om_{\mathbb{Z}}(X^*(\mathcal{T}), \mathbb{R}) = X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{R} \quad \text{and} \quad V_{\mathbb{C}} := \mathcal{H}om_{\mathbb{Z}}(X^*(\mathcal{T}), \mathbb{C}) = X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{C};$$

recall that there is an isomorphism $X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{C}^* \simeq \mathcal{T}$ given by $\lambda \otimes z \mapsto \lambda(z)$, therefore the map

$$\begin{array}{ccc} X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{C} & \longrightarrow & X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{C}^* \\ \lambda \otimes z & \longmapsto & \lambda \otimes e^{2\pi i z} \end{array}$$

induces a natural identification of $\mathcal{T}$ with ${}^{V_{\mathbb{C}}}/_{X_*(\mathcal{T})}$.

If we fix an isomorphism $\mathcal{T} \simeq (\mathbb{C}^*)^n$, we can work in concrete terms: the groups $X^*(\mathcal{T})$ and $X_*(\mathcal{T})$ are both identified with $\mathbb{Z}^n$, where

- an element $m = (a_1, \dots, a_n) \in \mathbb{Z}^n$ defines the character

$$\begin{array}{rccc} \chi_m\colon & (\mathbb{C}^*)^n & \longrightarrow & \mathbb{C}^* \\ & (t_1, \dots, t_n) & \longmapsto & t_1^{a_1} \cdots t_n^{a_n}; \end{array}$$

- an element $u = (b_1, \dots, b_n) \in \mathbb{Z}^n$ defines the one-parameter subgroup

$$\begin{array}{rccc} \lambda_u\colon & \mathbb{C}^* & \longrightarrow & (\mathbb{C}^*)^n \\ & t & \longmapsto & (t^{b_1}, \dots, t^{b_n}); \end{array}$$

moreover the pairing becomes

$$\begin{array}{rccc} \langle \cdot, \cdot \rangle\colon & \mathbb{Z}^n \times \mathbb{Z}^n & \longrightarrow & \mathbb{Z} \\ & (m, u) & \longmapsto & \sum_{i=1}^{n} a_i b_i. \end{array}$$

Definition 6.4. A *toric variety* is an irreducible variety X that contains an algebraic torus $\mathcal{T}$ as a Zariski open subset, and such that the action of $\mathcal{T}$ on itself extends to an algebraic action of $\mathcal{T}$ on X. (That means that the action map $\mathcal{T} \times X \to X$ is a morphism of algebraic varieties.)

If $\mathcal{T}$ is an algebraic complex torus, we say that X is a *$\mathcal{T}$-variety* if we want to stress that X contains $\mathcal{T}$ as an open subset.

We recall now the main definitions regarding polyhedral cones and fans, and how they relate to toric varieties.

Definition 6.5. Let $V = X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{R} \simeq \mathbb{R}^n$ as before. A *(convex polyhedral) cone* in V is a set of the form

$$C = C(\mathbf{r}_1, \ldots, \mathbf{r}_m) := \left\{ \sum_{j=1}^{m} \alpha_j \mathbf{r}_j \;\middle|\; \alpha_1 \geqslant 0, \ldots, \alpha_m \geqslant 0 \right\} \subseteq V$$

where $\mathbf{r}_1, \ldots, \mathbf{r}_m \in V$. We say that C is *generated* by $\mathbf{r}_1, \ldots, \mathbf{r}_m$. The cone $C(\mathbf{r}_1, \ldots, \mathbf{r}_m)$ is *rational* if $\mathbf{r}_1, \ldots, \mathbf{r}_m \in X_*(\mathcal{T}) \simeq \mathbb{Z}^n$.

Let $V^\vee$ be the dual space of V (which is isomorphic to $X^*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{R}$). Recall that the pairing $\langle \cdot, \cdot \rangle$ extends to

$$\begin{array}{rccc} \langle \cdot, \cdot \rangle_{\mathbb{R}}\colon & V^\vee \times V & \longrightarrow & \mathbb{R} \\ & (\chi \otimes \alpha, \lambda \otimes \beta) & \longmapsto & \langle \chi, \lambda \rangle \alpha\beta \end{array}$$

(we will drop the $\mathbb{R}$ subscript henceforth).

Definition 6.6. Given a cone $C \subseteq V$, its *dual cone* is defined by

$$C^\vee := \{\mathbf{f} \in V^\vee \mid \langle \mathbf{f}, \mathbf{r} \rangle \geqslant 0 \text{ for all } \mathbf{r} \in C\} \subseteq V^\vee.$$

Definition 6.7. For $\mathbf{f} \in V^\vee$, $\mathbf{f} \neq 0$, let $H_{\mathbf{f}} := \{\mathbf{r} \in V \mid \langle \mathbf{f}, \mathbf{r} \rangle = 0\} \subseteq V$. A *face* of a cone $C \subseteq V$ is a set of the form $H_{\mathbf{f}} \cap C$ for some $\mathbf{f} \in C^\vee$.

It is easy to prove that a face of a cone C is a polyhedral cone itself, that the intersection of two faces of C is again a face of C, and that a face of a face of C is again a face of C.

Definition 6.8. A *fan* Δ of V is a finite set of rational polyhedral cones in V such that

1. $\{0\}$ is a face of every $C \in \Delta$;
2. if D is a face of $C \in \Delta$, then $D \in \Delta$;
3. for every $C_1, C_2 \in \Delta$, $C_1 \cap C_2$ is a face of each (hence it belongs to Δ).

A *ray* of a fan Δ is a 1-dimensional cone $r \in \Delta$. We will sometimes use the term "ray" also to denote a generator of r, instead of the whole cone.

It is possible to associate a toric variety X_Δ with a fan Δ. In this construction each maximal cone $C \in \Delta$ describes a chart of X_Δ. An explicit description of it would require a long digression, which is beyond the scope of this work; the interested reader can find it in [16, Sections 1.2 and 3.1]. We recall only some properties of X_Δ that can be inferred from the fan Δ.

Definition 6.9. A rational polyhedral cone $C = C(\mathbf{r}_1, \dots, \mathbf{r}_m) \subseteq V$ is *smooth* if $(\mathbf{r}_1, \dots, \mathbf{r}_m)$ forms a part of a $\mathbb{Z}$-basis of $X_*(\mathcal{T})$. A fan Δ is *smooth* if every cone $C \in \Delta$ is smooth. A fan Δ is *complete* if

$$\bigcup_{C \in \Delta} C = V.$$

Proposition 6.10. *X_Δ is a smooth variety if and only if Δ is a smooth fan. X_Δ is compact (in the classical topology) if and only if Δ is a complete fan.*

Now that we listed the main basic definitions and properties of toric varieties, we get back to the torus itself, and define the analogue of a hyperplane arrangement in the torus version.

Definition 6.11. Let Γ be a split direct summand of $X^*(\mathcal{T})$, and let $\varphi \colon \Gamma \to \mathbb{C}^*$ be a homomorphism. A *layer* in $\mathcal{T}$ is the subvariety

$$\mathcal{K}(\Gamma, \varphi) \coloneqq \{t \in \mathcal{T} \mid \chi(t) = \varphi(\chi) \text{ for all } \chi \in \Gamma\}$$

Definition 6.12. A *toric arrangement* $\mathcal{A}$ is a finite set of layers $\{\mathcal{K}_1, \dots, \mathcal{K}_m\}$ in $\mathcal{T}$. A toric arrangement is called *divisorial* if $\operatorname{codim}(\mathcal{K}) = 1$ for all $\mathcal{K} \in \mathcal{A}$.

As we stated in the introduction to this chapter, in [12] it is shown how to build projective wonderful models for the complement

$$\mathcal{M}(\mathcal{A}) \coloneqq \mathcal{T} \setminus \bigcup_{\mathcal{K} \in \mathcal{A}} \mathcal{K}.$$

Definition 6.13. A *projective wonderful model* $Y_\mathcal{A}$ for $\mathcal{M}(\mathcal{A})$ is a smooth projective variety containing $\mathcal{M}(\mathcal{A})$ as a dense open set and such that the complement $Y_\mathcal{A} \setminus \mathcal{M}(\mathcal{A})$ is a divisor with normal crossings and smooth irreducible components.

Let $\mathcal{A} = \{\mathcal{K}_1, \dots, \mathcal{K}_r\}$ be a toric arrangement in the n-dimensional torus $\mathcal{T}$, where $\mathcal{K}_i = \mathcal{K}(\Gamma_i, \varphi_i)$ with Γ_i split direct summands of $X^*(\mathcal{T})$ and $\varphi_i \colon \Gamma_i \to \mathbb{C}^*$ homomorphisms, and let $V = X_*(\mathcal{T}) \underset{\mathbb{Z}}{\otimes} \mathbb{R}$ as before. Notice that a layer $\mathcal{K}(\Gamma, \varphi)$ is a coset with respect to the torus

$$H = \bigcap_{\chi \in \Gamma} \ker(\chi) \subseteq \mathcal{T}.$$

Definition 6.14. Let Δ be a fan in V. A character $\chi \in X^*(\mathcal{T})$ has the *equal sign property* with respect to Δ if, for every cone $C \in \Delta$, either $\langle \chi, \mathbf{c} \rangle \geqslant 0$ for all $\mathbf{c} \in C$ or $\langle \chi, \mathbf{c} \rangle \leqslant 0$ for all $\mathbf{c} \in C$.

Definition 6.15. Let Δ be a fan in V and let $\mathcal{K}(\Gamma, \varphi)$ be a layer. A $\mathbb{Z}$-basis $(\chi_1, \dots, \chi_m)$ for Γ is an *equal sign basis* with respect to Δ if χ_i has the equal sign property for all $i = 1, \dots, m$.

In [12, Proposition 6.1] it was shown how to construct, starting from a toric arrangement $\mathcal{A} = \{\mathcal{K}_1, \dots, \mathcal{K}_r\}$, a projective smooth $\mathcal{T}$-variety $X_{\mathcal{A}} := X_{\Delta(\mathcal{A})}$ such that for every layer $\mathcal{K}_i = \mathcal{K}(\Gamma_i, \varphi_i)$ in $\mathcal{A}$ there is an equal sign basis $(\chi_{i,1}, \dots, \chi_{i,s_i})$ of Γ_i with respect to the fan $\Delta(\mathcal{A})$. Given such a $\Delta(\mathcal{A})$, we will say that $X_{\mathcal{A}}$ is a *good toric variety* for $\mathcal{A}$. In the next section we describe the details of the algorithm that produces the fan $\Delta(\mathcal{A})$.

The behaviour of the layers in this variety $X_{\mathcal{A}}$ has been described in [12]. In fact, consider the closure $\overline{\mathcal{K}(\Gamma, \varphi)}$ of a layer in $X_{\mathcal{A}}$. It turns out that this closure is a toric variety itself, whose explicit description is provided by the following result.

Theorem 6.16 ([12, Proposition 3.1 and Theorem 3.1]). *For every layer $\mathcal{K}(\Gamma, \varphi)$ let H be the corresponding subtorus and let $V_\Gamma := \{\boldsymbol{v} \in V \mid \langle \chi, \boldsymbol{v} \rangle = 0$ for all $\chi \in \Gamma\}$.*

1. *For every cone $C \in \Delta(\mathcal{A})$, its relative interior is either entirely contained in V_Γ or disjoint from V_Γ.*
2. *The collection of cones $C \in \Delta(\mathcal{A})$ which are contained in V_Γ is a smooth fan $\Delta(\mathcal{A})_H$.*
3. *$\overline{\mathcal{K}(\Gamma, \varphi)}$ is a smooth H-variety whose fan is $\Delta(\mathcal{A})_H$.*
4. *Let $\mathcal{O}$ be an orbit of $\mathcal{T}$ in $X_{\mathcal{A}}$ and let $C_{\mathcal{O}} \in \Delta(\mathcal{A})$ be the corresponding cone. Then*
   - *(a) if $C_{\mathcal{O}}$ is not contained in V_Γ, $\overline{\mathcal{O}} \cap \overline{\mathcal{K}(\Gamma, \varphi)} = \varnothing$;*
   - *(b) If $C_{\mathcal{O}} \subset V_\Gamma$, $\mathcal{O} \cap \overline{\mathcal{K}(\Gamma, \varphi)}$ is the orbit of H in $\overline{\mathcal{K}(\Gamma, \varphi)}$ corresponding to $C_{\mathcal{O}} \in \Delta(\mathcal{A})_H$.*

Once we have the toric variety $X_{\mathcal{A}}$, the next step is to build the wonderful model. Let $\mathcal{Q}'$ be the set

$$\mathcal{Q}' := \{\overline{\mathcal{K}} \mid \mathcal{K} \in \mathcal{A}\}$$

and let

$$\mathcal{Q} := \mathcal{Q}' \cup \{D \mid D \text{ is an irreducible component of } X_{\mathcal{A}} \setminus \mathcal{T}\}.$$

As a consequence of Theorem 6.16, the family $\mathcal{L}$ of all the connected components of intersections of elements of $\mathcal{Q}$ gives an *arrangement of subvarieties* in the sense of Li's paper [32].

Definition 6.17. Let X be a non-singular variety. A *simple arrangement of subvarieties* of X is a finite set Λ of non-singular closed connected subvarieties properly contained in X such that

1. For every two $\Lambda_i, \Lambda_j \in \Lambda$, either $\Lambda_i \cap \Lambda_j \in \Lambda$ or $\Lambda_i \cap \Lambda_j = \varnothing$;
2. If $\Lambda_i \cap \Lambda_j \neq \varnothing$, the intersection is *clean*, i.e. it is non-singular and for every $y \in \Lambda_i \cap \Lambda_j$ we have the following conditions on the tangent spaces:

$$T_y(\Lambda_i \cap \Lambda_j) = T_y(\Lambda_i) \cap T_y(\Lambda_j).$$

Definition 6.18. Let X be a non-singular variety. An *arrangement of subvarieties* of X is a finite set Λ of non-singular closed connected subvarieties properly contained in X such that

1. For every two $\Lambda_i, \Lambda_j \in \Lambda$, either $\Lambda_i \cap \Lambda_j$ is a disjoint union of elements of Λ or $\Lambda_i \cap \Lambda_j = \varnothing$;
2. If $\Lambda_i \cap \Lambda_j \neq \varnothing$, the intersection is clean.

Notice that also the family $\mathcal{L}'$ of all the connected components of intersections of elements of $\mathcal{Q}'$ is an arrangement of subvarieties, because it is contained in $\mathcal{L}$ and it is closed under intersection. This allows, by a series of blow-ups, to build a projective wonderful model associated with $\mathcal{A}$.

6.2 Building the Fan

With the same notation as the one in the previous section, for each layer $\mathcal{K}_i = \mathcal{K}(\Gamma_i, \varphi_i)$ of $\mathcal{A}$, let $(\chi_{i,1}, \dots, \chi_{i,s_i})$ be a basis for Γ_i. We want to build a fan $\Delta(\mathcal{A})$ such that each $(\chi_{i,1}, \dots, \chi_{i,s_i})$ is an equal sign basis. To do so, we apply the following algorithm for all vectors in the set

$$\Xi := \bigcup_{\mathcal{K}_i \in \mathcal{A}} \{\chi_{i,1}, \dots, \chi_{i,s_i}\}. \tag{6.1}$$

```
def dcg_one_step(fan,vect):
    tempfan=copy(fan)
    while True: # the algorithm eventually stops anyway
        twocones=tempfan(2)
        bad=[]
        for cone in twocones:
            v1=vector(tuple(cone.ray(0)))
            v2=vector(tuple(cone.ray(1)))
```

```
9              if vect.dot_product(v1)*vect.dot_product(v2)<0:
10                 bad+=[cone]
11         if bad==[]:
12             return tempfan
13         else:
14             badlist=[]
15             for b in bad:
16                 v1=vector(tuple(b.ray(0)))
17                 v2=vector(tuple(b.ray(1)))
18                 val1=vect.dot_product(v1)
19                 val2=vect.dot_product(v2)
20                 if abs(val1)==abs(val2):
21                     badlist+=[(b,abs(val1),1)]
22                 else:
23                     badlist+=[(b,max(abs(val1),abs(val2)),0)]
24             c=sorted(badlist,key=lambda m:(m[1],m[2]))[-1][0]
25             newcones=[]
26             for conelist in tempfan.cones():
27                 for cone in conelist:
28                     if c.is_face_of(cone):
29                         oldrays=list(Set(cone.rays()).difference(Set(c.rays())))
30                         newcones+=[Cone(oldrays+[c.ray(0),c.ray(0)+c.ray(1)]),Cone(oldrays+[c.ray(1),c.ray(0)+c.ray(1)])]
31                     else:
32                         newcones+=[cone]
33             tempfan=Fan(newcones,discard_faces=True)
```

The inputs are a fan (fan) and a vector (vect) and the algorithm returns a new fan Δ, obtained by subdividing fan, such that for each cone $C \in \Delta$ we have either $\langle \texttt{vect}, \mathbf{c} \rangle \geqslant 0$ or $\langle \texttt{vect}, \mathbf{c} \rangle \leqslant 0$ for all $\mathbf{c} \in C$. The algorithm cycles indefinitely subdividing a fan tempfan that is equal to the input fan in the beginning (line 2) and then upgraded through the cycle.

In accordance with [12], we consider only the 2-dimensional cones (line 4): in fact, let $C \in \Delta$ be a k-dimensional cone generated by $(\mathbf{r}_1, \ldots, \mathbf{r}_k)$ and suppose that for each *2-dimensional* cone $C' \in \Delta$ we have either $\langle \texttt{vect}, \mathbf{c} \rangle \geqslant 0$ or $\langle \texttt{vect}, \mathbf{c} \rangle \leqslant 0$ for all $\mathbf{c} \in C'$. For each $\mathbf{r}_i$, $\mathbf{r}_j$ let $C(\mathbf{r}_i, \mathbf{r}_j) \in \Delta$ be the 2-dimensional cone generated by $\mathbf{r}_i$ and $\mathbf{r}_j$. Now, without loss of generality we may assume that $\langle \texttt{vect}, \mathbf{r}_1 \rangle \geqslant 0$ and $\langle \texttt{vect}, \mathbf{r}_2 \rangle \geqslant 0$, since we have the property for $C(\mathbf{r}_1, \mathbf{r}_2)$. But now also $\langle \texttt{vect}, \mathbf{r}_3 \rangle \geqslant 0$, due to the property applied to the cone $C(\mathbf{r}_2, \mathbf{r}_3)$. By induction then we have $\langle \texttt{vect}, \mathbf{r}_i \rangle \geqslant 0$ for all $i = 1, \ldots, k$.

Notice that for 2-dimensional cones of the form $C(v_1, v_2)$ the property translates as

$$\langle \texttt{vect}, v_1 \rangle \langle \texttt{vect}, v_2 \rangle \geqslant 0, \tag{6.2}$$

and we check it for each 2-dimensional cone in `tempfan`, building a list of "bad" cones for which the property is not satisfied (lines 5–10). If there are no bad cones, the algorithm terminates returning the fan (lines 11–12); otherwise, we choose a bad cone $C = C(v_1, v_2)$ and define a new fan $\Delta(C)$ obtained from $\Delta =$ `tempfan` by substituting each cone $C(v_1, v_2, w_1, \dots, w_k)$ containing C with two new cones generated by $(v_1, v_1 + v_2, w_1, \dots, w_k)$ and $(v_1 + v_2, v_2, w_1, \dots, w_k)$ respectively.

Proposition 6.19 ([12, Proposition 4.1]). *The new fan $\Delta(C)$ is smooth, and a proper subdivision of Δ. Moreover, if X_Δ and $X_{\Delta(C)}$ are the two toric varieties associated with the fans Δ and $\Delta(C)$ respectively, then $X_{\Delta(C)}$ is obtained from X_Δ by blowing up the closure of the 2-codimensional orbit in X_Δ associated with C.*

The only thing to do is to find a way to choose wisely the bad cone that has to be replaced. We follow the choice of [12]: if $\Delta_{(N)}$ is the set of the bad 2-dimensional cones, define

$$\begin{array}{rccc} P_\Delta\colon & \Delta_{(N)} & \longrightarrow & \mathbb{N} \times \{0,1\} \\ & C(v_1, v_2) & \longmapsto & (M_C, \varepsilon_C) \end{array}$$

where $M_C = \max\{|\langle \texttt{vect}, v_1 \rangle|, |\langle \texttt{vect}, v_2 \rangle|\}$ and

$$\varepsilon_C = \begin{cases} 1 & \text{if } |\langle \texttt{vect}, v_1 \rangle| = |\langle \texttt{vect}, v_2 \rangle| \\ 0 & \text{otherwise.} \end{cases}$$

Fix the lexicographic order on $\mathbb{N} \times \{0,1\}$, i.e.

$$(0,0) < (0,1) < (1,0) < (1,1) < (2,0) < \cdots$$

Lemma 6.20 ([12, Lemma 4.2]). *Assume $\Delta_{(N)} \neq \varnothing$ and choose $C \in \Delta_{(N)}$ such that $P_\Delta(C) = (M_C, \varepsilon_C)$ is maximum in $\mathrm{Im}(P_\Delta)$.*

1. *If $\varepsilon_C = 1$, then $\Delta(C)_{(N)} = \Delta_{(N)} \smallsetminus \{C\}$.*
2. *If $\varepsilon_C = 0$, then $\max(\mathrm{Im}(P_{\Delta(C)})) \leqslant (M_C, \varepsilon_C)$, and*

$$\#\Big(P^{-1}_{\Delta(C)}((M_C, \varepsilon_C))\Big) < \#\big(P^{-1}_{\Delta}((M_C, \varepsilon_C))\big).$$

The previous lemma proves that, if we choose $C \in \Delta_{(N)}$ such that $P_\Delta(C) = (M_C, \varepsilon_C)$ is maximum in $\mathrm{Im}(P_\Delta)$, we are guaranteed that the number of bad cones eventually decreases and the algorithm stops.

Lines 14–24 choose this maximum cone: in particular, `badlist` contains the graph of the map P_Δ, that is to say, it is the list of triples

$$\left[(C, M_C, \varepsilon_C) \in \Delta_{(N)} \times \mathbb{N} \times \{0,1\} \,\middle|\, C \in \Delta_{(N)}\right].$$

This list is sorted lexicographically with respect to the order of $\mathbb{N} \times \{0,1\}$, and the first component of the last triple is taken and stored in the variable `c` (line 24). At last we build the new fan (lines 25–33): for each cone C of `tempfan`, if `c` is a face of C, take the generators of C and add the two new cones to `newcones`; otherwise, just leave C as it is.

The algorithm `dcg_one_step` allows us to prove the following statement.

Proposition 6.21 (see [12, Proposition 6.1]). *If $\mathcal{A}$ is a toric arrangement in the torus $\mathcal{T}$, then there exists a fan $\Delta(\mathcal{A})$ such that*

1. *$X_{\Delta(\mathcal{A})}$ is a smooth $\mathcal{T}$-variety, obtained from $(\mathbb{P}^1)^n$ by a sequence of blow-ups along closures of 2-codimensional orbits;*

2. *every layer $\mathcal{K}_i \in \mathcal{A}$ has an equal sign basis with respect to $\Delta(\mathcal{A})$.*

Proof. Apply the algorithm `dcg_algorithm` with inputs `vectors` $= \Xi$ defined in (6.1), and the fan Ω induced by the decomposition in orthants of $\mathbb{R}^n$ (recall that the associated toric variety X_Ω is $(\mathbb{P}^1)^n$). □

```
def dcg_algorithm(fan,vectors):
    tempfan=copy(fan)
    for v in vectors:
        tempfan=dcg_one_step(tempfan,v)
    return tempfan
```

Remark. Notice that the construction of the fan $\Delta(\mathcal{A})$ depends on the choice of several parameters:

- a basis for the characters group $X^*(\mathcal{T})$, so to identify a character with a vector in $\mathbb{Z}^n$;
- the set of characters Ξ;
- the initial input fan, in this case the orthant fan Ω.

Moreover, the set Ξ depends on the choice of a basis of Γ_i for each layer $\mathcal{K}_i = \mathcal{K}(\Gamma_i, \varphi_i)$ of $\mathcal{A}$. In the next section we see another algorithm that works in dimension 2 and does not need an input fan, thus removing a layer of arbitrariness in the construction of $\Delta(\mathcal{A})$. Furthermore the fact that the ambient space is 2-dimensional forces the choice of a basis for Γ_i: if $\mathcal{K}_i$ is 1-dimensional, then there is a unique (up to sign) primitive vector $\boldsymbol{v}_i$ such that $\Gamma_i = \langle \boldsymbol{v}_i \rangle$, and if $\mathcal{K}_i$ is 0-dimensional, then we may choose $\Gamma_i = \langle (1,0),(0,1) \rangle$.

6.3 Another Way to Build the Fan

The algorithm described in the previous section begins with a smooth fan and subdivides it in such a way that:

- each intermediate subdivision of the fan remains smooth;
- in the end we have a fan such that each vector of the set Ξ has the equal sign property with respect to that fan.

Now notice that, given a set of vectors Ξ, there is a "canonical" fan with respect to which each vector of Ξ has the equal sign property: it is the one induced by the hyperplanes orthogonal to the vectors of Ξ. The problem is that in general this fan is not smooth; the following algorithm subdivides it so that in the end we get a smooth fan, in the 2-dimensional case.

```
def smooth_equal_sign_fan(veclist):
    startrays=sorted([positive_orthogonal(v) for v in veclist],key=lambda v: div_infty(v[0],v[1]),reverse=True)
    startrays+=[-startrays[0]]
    finalrays=[]
    for i in xrange(len(startrays)-1):
        v1=startrays[i]
        v2=startrays[i+1]
        finalrays+=[v1] # the first ray of the cone is untouched
        p=abs(v1[0]*v2[1]-v2[0]*v1[1]) # =|Det([v1|v2])|
        while p!=1:
            c1,c2=xgcd(v1[0],v1[1])[1:3] # xgcd(a,b) computes (g,c1,c2) where g=GCD(a,b) and g=c1*a+c2*b
            q=c1*v2[0]+c2*v2[1]
            q0=q%p
            newv=vector([v1[0]*(p+q-q0)//p-c2,v1[1]*(p+q-q0)//p+c1])
            finalrays+=[newv]
            v1=newv
            p=abs(v1[0]*v2[1]-v2[0]*v1[1])
    finalrays+=[-v for v in finalrays]
    cones=[]
    for i in xrange(len(finalrays)-1):
        cones+=[Cone([finalrays[i],finalrays[i+1]])]
    cones+=[Cone([finalrays[-1],finalrays[0]])]
    return Fan(cones)
```

Notice that this time the input is just the set $\Xi = \texttt{veclist}$,[1] because the starting fan is defined from it: in fact, we compute its rays in line 2. Here we use the auxiliary function `positive_orthogonal`, that takes a 2-dimensional vector $v = (v_1, v_2)$ and returns either $(v_2, -v_1)$ or $(-v_2, v_1)$ depending on which one belongs to the upper half-plane. These rays are sorted counter-clockwise starting from the positive x-semiaxis; to do so, we defined a function

$$\texttt{div\_infty(a,b)} := \begin{cases} a/b & \text{if } b \neq 0 \\ \operatorname{sgn}(a) \cdot \infty & \text{otherwise.} \end{cases}$$

We then add the opposite of the first vector, so that `startrays` contains all the rays delimiting the "upper cones" of the fan (see Figure 6.1).

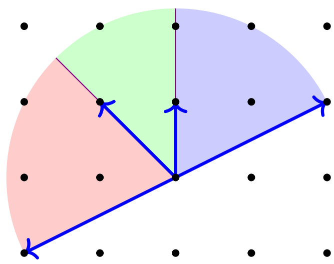


Figure 6.1: Rays in the list `startrays` for `veclist` $= \{(-1,0), (1,1), (1,-2)\}$, with the cones highlighted. Dots represent the lattice points.

The lines 5–17 divide each cone so that the resulting fan is smooth. Suppose that the two rays delimiting a cone C are $\texttt{v1} = (x, y)$ and $\texttt{v2} = (z, w)$; smoothness is guaranteed as long as

$$\det \begin{pmatrix} x & z \\ y & w \end{pmatrix} = \pm 1,$$

so we compute it (or better, its absolute value) and we call it $\texttt{p}$ (line 8). If $\texttt{p} = 1$, we leave the cone untouched; otherwise, we proceed in the following way. Let `c1`, `c2` such that $\texttt{c1}\,x + \texttt{c2}\,y = 1$ (they exist because `v1` is supposed to be primitive; line 11) and notice that the value $\texttt{c1}\,z + \texttt{c2}\,w \pmod{\texttt{p}}$ does not depend on the choice of `c1` and `c2`. In fact, let c_1', c_2' be another such choice; therefore

$$0 = 1 - 1 = (\texttt{c1}\,x + \texttt{c2}\,y) - (c_1' x + c_2' y) = (\texttt{c1} - c_1')x + (\texttt{c2} - c_2')y$$

and on the other hand there exists k such that

$$\begin{pmatrix} z \bmod \texttt{p} \\ w \bmod \texttt{p} \end{pmatrix} = k \begin{pmatrix} x \bmod \texttt{p} \\ y \bmod \texttt{p} \end{pmatrix}$$

[1] It is assumed that Ξ contains only primitive vectors and does not contain parallel vectors.

(this is because $\det \equiv 0 \pmod{\mathtt{p}}$, but none of the vectors can be the zero vector modulo $\mathtt{p}$ since they are primitive). It follows that

$$\begin{aligned}(\mathtt{c1}\,z + \mathtt{c2}\,w) - (c_1' z + c_2' w) &= (\mathtt{c1} - c_1')z + (\mathtt{c2} - c_2')w \\ &\equiv k\big((\mathtt{c1} - c_1')x + (\mathtt{c2} - c_2')y\big) \equiv 0 \pmod{\mathtt{p}}.\end{aligned}$$

Now let $\mathtt{q} = \mathtt{c1}\,z + \mathtt{c2}\,w$ and let $\mathtt{q0}$ be the remainder of the division of $\mathtt{q}$ by $\mathtt{p}$ (lines 12 and 13). Notice that $0 \leqslant \mathtt{q0} < \mathtt{p}$ and that $\mathrm{GCD}(\mathtt{q0}, \mathtt{p}) = 1$ (by absurd: let $\mathrm{GCD}(\mathtt{q0}, \mathtt{p}) = h > 1$; since $h \mid \mathtt{q0}$ and $h \mid \mathtt{p}$, by definition $h \mid \mathtt{q}$, hence the vector $(\mathtt{q}, \mathtt{p})$ is not primitive; but

$$\begin{pmatrix} \mathtt{c1} & \mathtt{c2} \\ -y & x \end{pmatrix} \begin{pmatrix} z \\ w \end{pmatrix} = \begin{pmatrix} \mathtt{q} \\ \mathtt{p} \end{pmatrix}$$

and the determinant of the matrix is $\mathtt{c1}\,x + \mathtt{c2}\,y = 1$, so it sends primitive vectors to primitive vectors). Therefore we define the vector (line 14)

$$\mathtt{newv} := \frac{1}{\mathtt{p}}\big((\mathtt{p} - \mathtt{q0})\mathtt{v1} + \mathtt{v2}\big) = \frac{1}{\mathtt{p}} \begin{pmatrix} (\mathtt{p} + \mathtt{q} - \mathtt{q0})x - \mathtt{c2}\,\mathtt{p} \\ (\mathtt{p} + \mathtt{q} - \mathtt{q0})y + \mathtt{c1}\,\mathtt{p} \end{pmatrix},$$

where it is an easy check that the division is exact, i.e. $\mathtt{newv} \in \mathbb{Z}^2$. Now notice that $\mathtt{newv}$ belongs to the cone generated by $\mathtt{v1}$ and $\mathtt{v2}$, because it is a linear combination of them with positive coefficients (remember that $\mathtt{q0} < \mathtt{p}$); moreover,

$$\det\left(\mathtt{v1} \,\middle|\, \mathtt{newv}\right) = \frac{\mathtt{p} - \mathtt{q0}}{\mathtt{p}} \det\left(\mathtt{v1} \,\middle|\, \mathtt{v1}\right) + \frac{1}{\mathtt{p}} \det\left(\mathtt{v1} \,\middle|\, \mathtt{v2}\right) = \pm\frac{\mathtt{p}}{\mathtt{p}} = \pm 1$$

so the cone $C(\mathtt{v1}, \mathtt{newv})$ is smooth. On the other hand

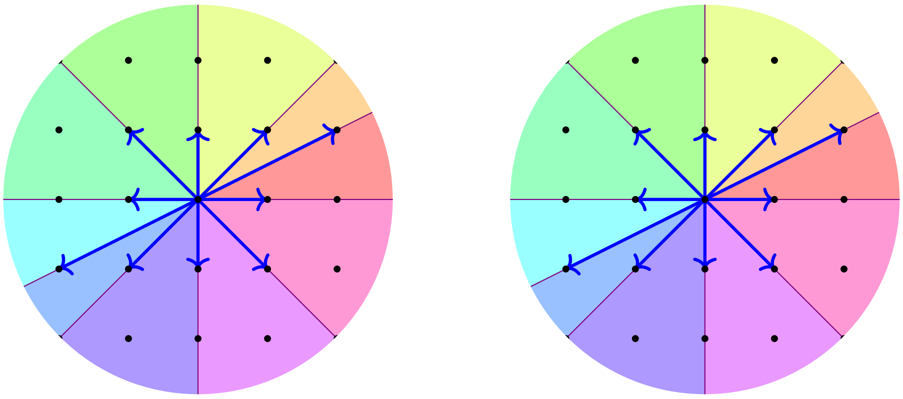
$$\det\left(\mathtt{newv} \,\middle|\, \mathtt{v2}\right) = \frac{\mathtt{p} - \mathtt{q0}}{\mathtt{p}} \det\left(\mathtt{v1} \,\middle|\, \mathtt{v2}\right) + \frac{1}{\mathtt{p}} \det\left(\mathtt{v2} \,\middle|\, \mathtt{v2}\right) = \pm\frac{\mathtt{p}(\mathtt{p} - \mathtt{q0})}{\mathtt{p}} = \pm(\mathtt{p} - \mathtt{q0})$$

and $|\mathtt{p} - \mathtt{q0}| < \mathtt{p}$, so we can reapply the algorithm to the cone $C(\mathtt{newv}, \mathtt{v2})$ (lines 15–17). Since the absolute value of the new determinant strictly decreases, we can prove by induction that the algorithm terminates with $\mathtt{p} = 1$.

Once all the (upper) cones have been subdivided, we add all the opposites of the rays to the list `finalrays` (line 18), compute the actual cones (lines 19–22) and return the fan.

6.4 Examples

In this section we show some 2-dimensional fans computed with the two algorithms analysed in the previous sections. In the figures, the fan on the left is obtained with the `dcg_algorithm` algorithm, and the fan on the right with the `smooth_equal_sign_fan` one.



(a) Output of `dcg_algorithm`. (b) Output of `smooth_equal_sign_fan`.

Figure 6.2: Fans starting from $\Xi = \{(-1,0),(1,1),(1,-2)\}$.

Example 6.1. The first example is obtained from $\Xi = \{(-1,0),(1,1),(1,-2)\}$ (Figure 6.2). For this example we detail each step of the two algorithms; let us begin with `dcg_algorithm`.

The starting fan is the orthant fan Ω, whose 2-dimensional cones are $C(\boldsymbol{e}_1,\boldsymbol{e}_2)$, $C(\boldsymbol{e}_2,-\boldsymbol{e}_1)$, $C(-\boldsymbol{e}_1,-\boldsymbol{e}_2)$ and $C(-\boldsymbol{e}_2,\boldsymbol{e}_1)$, where $\boldsymbol{e}_1 = (1,0)$ and $\boldsymbol{e}_2 = (0,1)$. The first vector of Ξ is $\boldsymbol{v}_1 = (-1,0)$; let us look for the bad cones:

$$\begin{aligned} \langle \boldsymbol{v}_1, \boldsymbol{e}_1 \rangle &= -1, & \langle \boldsymbol{v}_1, \boldsymbol{e}_2 \rangle &= 0, \\ \langle \boldsymbol{v}_1, -\boldsymbol{e}_1 \rangle &= 1, & \langle \boldsymbol{v}_1, -\boldsymbol{e}_2 \rangle &= 0. \end{aligned}$$

Since all the cones of Ω contain one between $\boldsymbol{e}_2$ and $-\boldsymbol{e}_2$ as a generator, condition (6.2) is always satisfied, because

$$\langle \boldsymbol{v}_1, \pm\boldsymbol{e}_1 \rangle \langle \boldsymbol{v}_1, \pm\boldsymbol{e}_2 \rangle = 0.$$

Therefore $\boldsymbol{v}_1$ already has the equal sign property with respect to Ω, and the fan is untouched. The algorithm proceeds with the next vector of Ξ, that is $\boldsymbol{v}_2 = (1,1)$. This time we have

$$\begin{aligned} \langle \boldsymbol{v}_2, \boldsymbol{e}_1 \rangle &= 1, & \langle \boldsymbol{v}_2, \boldsymbol{e}_2 \rangle &= 1, \\ \langle \boldsymbol{v}_2, -\boldsymbol{e}_1 \rangle &= -1, & \langle \boldsymbol{v}_2, -\boldsymbol{e}_2 \rangle &= -1, \end{aligned}$$

so there are two bad cones: $C_1 = C(\boldsymbol{e}_2,-\boldsymbol{e}_1)$ and $C_2 = C(-\boldsymbol{e}_2,\boldsymbol{e}_1)$. We apply P_Ω to choose the one to subdivide:

$$P_\Omega(C_1) = P_\Omega(C_2) = (1,1),$$

therefore the algorithm just chooses one of them.[▸2] In the subdivision part of the

[▸2]This actually depends on how the `sorted` function is implemented in `SageMath`; for this example we suppose that in line 24 the algorithm sets $c = C_1$.

algorithm, the new cones are

$$C(\boldsymbol{e}_2, \boldsymbol{e}_2 - \boldsymbol{e}_1) \text{ and } C(\boldsymbol{e}_2 - \boldsymbol{e}_1, -\boldsymbol{e}_1)$$

so the new fan $\Delta_{(1)}$ has now five 2-dimensional cones. The algorithm now keeps the vector $\boldsymbol{v}_2$ and checks the property with the new fan: since

$$\langle \boldsymbol{v}_2, \boldsymbol{e}_2 - \boldsymbol{e}_1 \rangle = \langle \boldsymbol{v}_2, \boldsymbol{e}_2 \rangle - \langle \boldsymbol{v}_2, \boldsymbol{e}_1 \rangle = 0,$$

the two new cones of $\Delta_{(1)}$ satisfy the equal sign property, and the only bad cone is C_2. The algorithm subdivides it in

$$C(-\boldsymbol{e}_2, \boldsymbol{e}_1 - \boldsymbol{e}_2) \text{ and } C(\boldsymbol{e}_1 - \boldsymbol{e}_2, \boldsymbol{e}_1)$$

so that the new fan $\Delta_{(2)}$ has now six 2-dimensional cones. A quick computation shows that now $\boldsymbol{v}_2$ has the equal sign property with respect to $\Delta_{(2)}$, so the algorithm proceeds with the final vector of Ξ, $\boldsymbol{v}_3 = (1, -2)$. We compute all the products

$$\begin{array}{cc} \langle \boldsymbol{v}_3, \boldsymbol{e}_1 \rangle = 1, & \langle \boldsymbol{v}_3, \boldsymbol{e}_2 \rangle = -2, \\ \langle \boldsymbol{v}_3, -\boldsymbol{e}_1 \rangle = -1, & \langle \boldsymbol{v}_3, -\boldsymbol{e}_2 \rangle = 2, \\ \langle \boldsymbol{v}_3, \boldsymbol{e}_2 - \boldsymbol{e}_1 \rangle = -3, & \langle \boldsymbol{v}_3, \boldsymbol{e}_1 - \boldsymbol{e}_2 \rangle = 3, \end{array}$$

and discover that the bad cones of $\Delta_{(2)}$ are $C'_1 = C(\boldsymbol{e}_1, \boldsymbol{e}_2)$ and $C'_2 = C(-\boldsymbol{e}_1, -\boldsymbol{e}_2)$. Therefore the algorithm computes

$$P_{\Delta_{(2)}}(C'_1) = P_{\Delta_{(2)}}(C'_2) = (2, 0),$$

and we suppose that it chooses to subdivide C'_1, producing the new fan $\Delta_{(3)}$ with the two new cones

$$C'_{1,1} = C(\boldsymbol{e}_1, \boldsymbol{e}_1 + \boldsymbol{e}_2) \text{ and } C'_{1,2} = C(\boldsymbol{e}_1 + \boldsymbol{e}_2, \boldsymbol{e}_2).$$

Now, $\langle \boldsymbol{v}_3, \boldsymbol{e}_1 + \boldsymbol{e}_2 \rangle = -1$, therefore the bad cones of $\Delta_{(3)}$ are C'_2 and $C'_{1,1}$. Since

$$P_{\Delta_{(3)}}(C'_2) = (2, 0) > (1, 1) = P_{\Delta_{(3)}}(C'_{1,1}),$$

the algorithm subdivides C'_2 in

$$C'_{2,1} = C(-\boldsymbol{e}_1, -\boldsymbol{e}_1 - \boldsymbol{e}_2) \text{ and } C'_{2,2} = C(-\boldsymbol{e}_1 - \boldsymbol{e}_2, -\boldsymbol{e}_2),$$

producing the fan $\Delta_{(4)}$. We compute $\langle \boldsymbol{v}_3, -\boldsymbol{e}_1 - \boldsymbol{e}_2 \rangle = 1$ and find that the bad cones of $\Delta_{(4)}$ are $C'_{1,1}$ and $C'_{2,1}$. The function $P_{\Delta_{(4)}}$ assumes the same value on them, which is $(1, 1)$, so we suppose that the algorithm subdivides $C'_{1,1}$ in

$$C(\boldsymbol{e}_1, 2\boldsymbol{e}_1 + \boldsymbol{e}_2) \text{ and } C(2\boldsymbol{e}_1 + \boldsymbol{e}_2, \boldsymbol{e}_1 + \boldsymbol{e}_2).$$

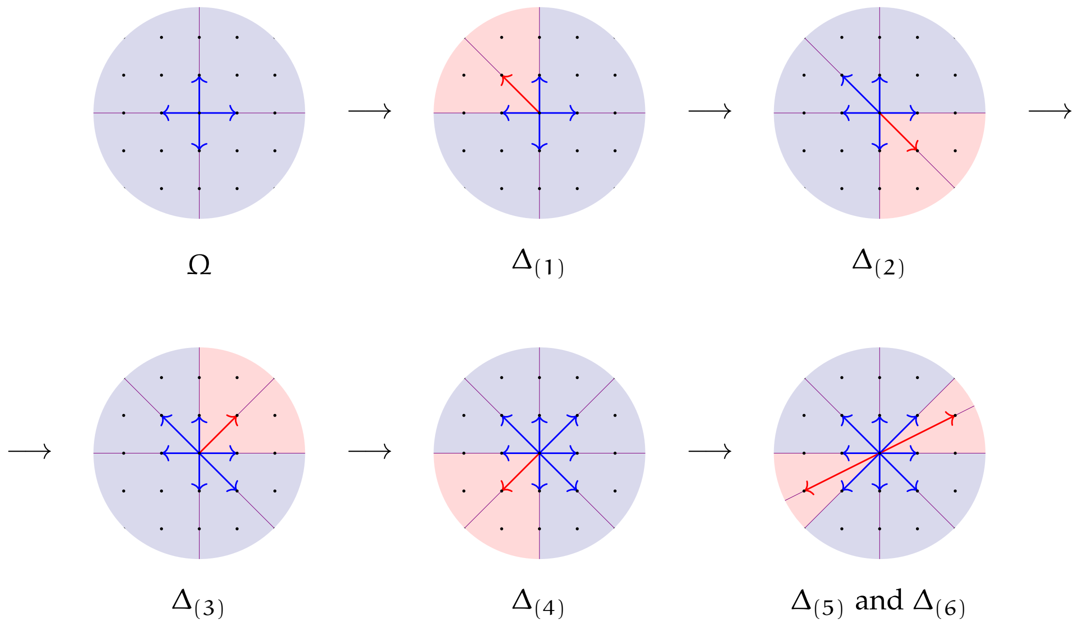


Figure 6.3: Series of steps of `dcg_algorithm` to create the fan of Figure 6.2a.

Finally we get $\langle v_3, 2e_1 + e_2 \rangle = 0$, therefore the only bad cone of the new fan $\Delta_{(5)}$ is $C'_{2,1}$, which is subdivided in

$$C(-e_1, -2e_1 - e_2) \text{ and } C(-2e_1 - e_2, -e_1 - e_2).$$

This last fan $\Delta_{(6)}$ has no bad cones and the algorithm terminates.

Now let's look at the `smooth_equal_sign_fan` algorithm. At first it computes the positive orthogonal vectors, which are $w_1 = (0, 1)$, $w_2 = (-1, 1)$ and $w_3 = (2, 1)$. The three upper cones, ordered counter-clockwise from the positive x-axis, are

$$C_1 = C(w_3, w_1), \quad C_2 = C(w_1, w_2), \quad C_3 = C(w_2, -w_3).$$

For each cone, now, the algorithm checks if it is smooth. The first one is C_1:

$$\det \left(w_3 \;\middle|\; w_1 \right) = \det \begin{pmatrix} 2 & 0 \\ 1 & 1 \end{pmatrix} = 2$$

so the cone has to be subdivided. The vector w_3 is added to `finalrays` and all the coefficients are computed: $c_1 = 1$ and $c_2 = -1$ (because $1 \cdot 2 + (-1) \cdot 1 = 1$), therefore $q = 1 \cdot 0 + (-1) \cdot 1 = -1$ and $q_0 = -1 \bmod 2 = 1$. The new ray is

$$w'_1 = \frac{1}{2}\big((2-1)w_3 + w_1\big) = \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

which is added to `finalrays`. The cone $C(w_3, w'_1)$ is guaranteed to be smooth: the algorithm checks $C(w'_1, w_1)$ computing

$$\det\left(w'_1 \,\middle|\, w_1\right) = \det\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} = 1.$$

Also this cone is smooth; the algorithm then proceeds with the second cone C_2. It turns out that

$$\det\left(w_1 \,\middle|\, w_2\right) = \det\begin{pmatrix} 0 & -1 \\ 1 & 1 \end{pmatrix} = 1$$

so C_2 is already smooth and it is left untouched. The last cone to test is C_3:

$$\det\left(w_2 \,\middle|\, -w_3\right) = \det\begin{pmatrix} -1 & -2 \\ 1 & -1 \end{pmatrix} = 3$$

so the algorithm has to subdivide the cone. It computes all the numbers it needs: $c_1 = 0$ and $c_2 = 1$ is a possible choice that brings $q = 0 \cdot (-2) + 1 \cdot (-1) = -1$ and $q_0 = -1 \bmod 3 = 2$. Therefore the new vector is

$$w'_2 = \frac{1}{3}((3-2)w_2 - w_3) = \begin{pmatrix} -1 \\ 0 \end{pmatrix}.$$

As before, the cone $C(w_2, w'_2)$ is smooth; for the other cone $C(w'_2, -w_3)$ the algorithm computes

$$\det\left(w'_2 \,\middle|\, -w_3\right) = \det\begin{pmatrix} -1 & -2 \\ 0 & -1 \end{pmatrix} = 1$$

so no more subdivision is needed. Finally the algorithm adds all the negative vectors and returns the fan pictured in Figure 6.2b.

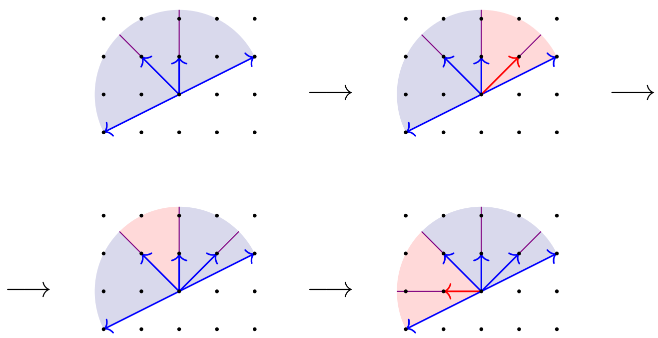
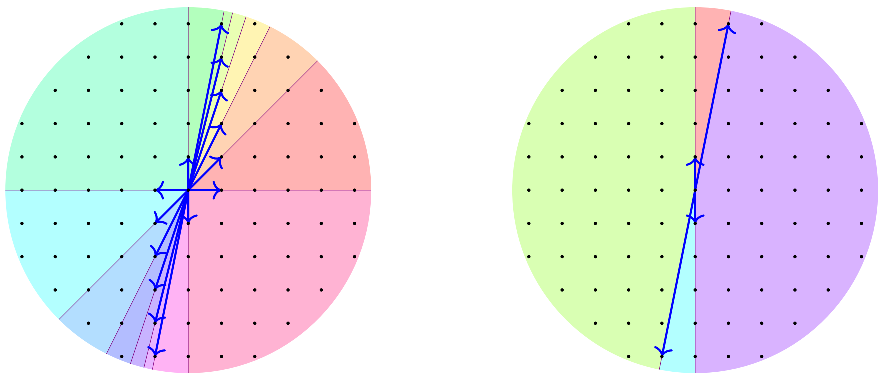


Figure 6.4: Series of steps of `smooth_equal_sign_fan` that divides the upper cones.



(a) Output of `dcg_algorithm`. (b) Output of `smooth_equal_sign_fan`.

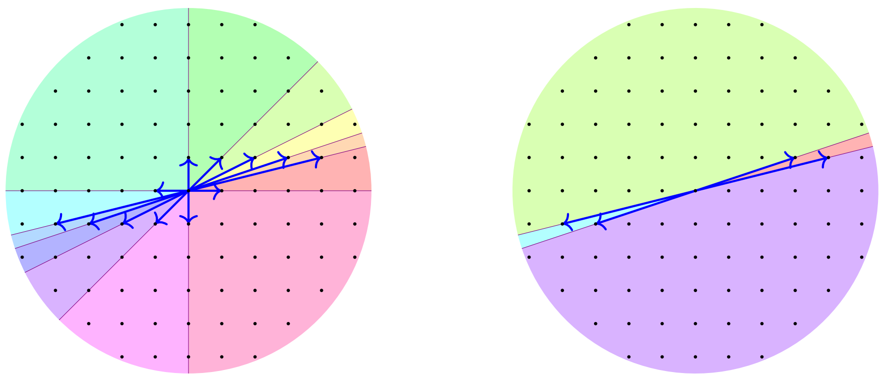
Figure 6.5: Fans starting from $\Xi = \{(1,0),(5,-1)\}$.

The previous example seems to suggest that the two algorithms return the same fan. This is not true in general, as the following example shows.

Example 6.2. Consider the two outputs for the set of vectors $\Xi = \{(1,0),(5,-1)\}$. The fan of Figure 6.5b has just four cones, that are the ones generated by the rays orthogonal to the vectors in Ξ. This is because these cones are already smooth: for example, if we focus on $C\big((1,5),(0,1)\big)$, we have

$$\det\begin{pmatrix} 1 & 0 \\ 5 & 1 \end{pmatrix} = 1.$$

On the other hand, `dcg_algorithm` adds the vector $(1,0)$ in the fan, which not only has nothing to do with the vectors of Ξ, but also forces the algorithm to produce a great amount of other vectors. A similar situation happens in the fans of Figure 6.6.



(a) Output of `dcg_algorithm`. (b) Output of `smooth_equal_sign_fan`.

Figure 6.6: Fans starting from $\Xi = \{(-1,3),(-1,4)\}$.

Chapter 7

Cohomology of the Wonderful Model

In the previous chapter we have seen how it is possible to construct a projective wonderful model $Y_{\mathcal{A}}$ for the complement of a toric arrangement $\mathcal{A}$. To do so, we defined a good toric variety $X_{\mathcal{A}}$ by subdividing suitably a given fan. Now we would like to know something more about the topology of $Y_{\mathcal{A}}$. We follow a recent article by De Concini and Gaiffi [11], where it is described an explicit presentation of $H^*(Y_{\mathcal{A}};\mathbb{Z})$ as a quotient of a polynomial ring with coefficients in $H^*(X_{\mathcal{A}};\mathbb{Z})$.

The cohomology ring $H^*(X_{\mathcal{A}};\mathbb{Z})$ can be described in terms of the fan associated with $X_{\mathcal{A}}$ (see Theorem 7.2 below), so we are able to compute it. The other main objects involved in the description of the presentation of $H^*(Y_{\mathcal{A}};\mathbb{Z})$ are the poset of layers $\mathcal{C}(\mathcal{A})$ (Definition 7.1) and a well-connected building set (Definition 7.5). It is not difficult to put this information together and develop an algorithm that computes the presentation of $H^*(Y_{\mathcal{A}};\mathbb{Z})$. In this chapter we outline an algorithm that does so and give some examples of cohomology rings of wonderful models obtained through it.

7.1 A Presentation of the Cohomology Ring

The first ingredient needed to compute the cohomology ring is the "toric version" of the intersection poset of a hyperplane arrangement.

Definition 7.1. Let $\mathcal{A}$ be a toric arrangement in the torus $\mathcal{T}$. The *poset of layers* $\mathcal{C}(\mathcal{A})$ is the set of all the connected components of the non-empty intersections of the layers of $\mathcal{A}$, partially ordered by reverse inclusion. It includes $\mathcal{T}$ as the intersection of zero layers.

Notice that for toric arrangements the intersection of two layers may not be connected, but each connected component is a layer. In particular, if $\mathcal{K}_1 = \mathcal{K}(\Gamma_1, \varphi_1)$ and

$\mathcal{K}_2 = \mathcal{K}(\Gamma_2, \varphi_2)$, then each connected component of $\mathcal{K}_1 \cap \mathcal{K}_2$ is a layer $\mathcal{K}(\Gamma, \varphi)$ where Γ is the saturation of $\Gamma_1 + \Gamma_2$.[1]

Example 7.1. Let $\mathcal{A} = \{\mathcal{K}_1, \mathcal{K}_2\} \subseteq (\mathbb{C}^*)^2$ be the toric arrangement where $\mathcal{K}_i = \mathcal{K}(\Gamma_i, \varphi_i)$ with

$$\Gamma_1 = \left\langle \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\rangle, \ \varphi_1 \colon \begin{pmatrix} 1 \\ 0 \end{pmatrix} \mapsto 1 \qquad \text{and} \qquad \Gamma_2 = \left\langle \begin{pmatrix} 1 \\ 2 \end{pmatrix} \right\rangle, \ \varphi_2 \colon \begin{pmatrix} 1 \\ 2 \end{pmatrix} \mapsto 1,$$

in other words $\mathcal{K}_1 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 = 1\}$ and $\mathcal{K}_2 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 t_2^2 = 1\}$. It is immediate to show that

$$\mathcal{K}_1 \cap \mathcal{K}_2 = \{(1, 1), (1, -1)\},$$

therefore the Hasse diagram of $\mathcal{C}(\mathcal{A})$ has the form pictured in Figure 7.1.

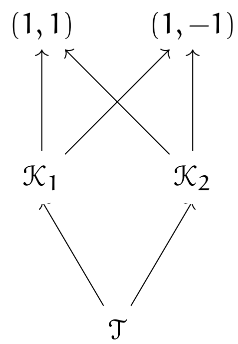


Figure 7.1: Hasse diagram of $\mathcal{C}(\mathcal{A})$ for the arrangement of Example 7.1.

In order to compute the poset of layers of a toric arrangement $\mathcal{A}$, we will use an algorithm outlined by Matthias Lenz in [31]. However, his definition of toric arrangements is slightly different from ours: first of all, Lenz deals with the *real compact* torus $(S^1)^n$ instead of the complex algebraic torus $(\mathbb{C}^*)^n$. In Lenz's view, a toric arrangement is a finite collection of (possibly disconnected) hypersurfaces

$$\mathcal{A} = \{S_1, \ldots, S_m\} \subseteq (S^1)^n$$

where S_i is defined by a vector $\boldsymbol{v}_i = (v_{i,1}, \ldots, v_{i,n}) \in \mathbb{Z}^n$ in the following way:

$$S_i \coloneqq \{(\alpha_1, \ldots, \alpha_n) \in (S^1)^n \mid \alpha_1^{v_{i,1}} \cdots \alpha_n^{v_{i,n}} = 1\}. \tag{7.1}$$

The vectors $\boldsymbol{v}_i$ are *not* required to be primitive; in particular, if $\mathrm{GCD}(v_{i,1}, \ldots, v_{i,n}) = d$, then S_i has d connected components.

[1] Recall that, if $L \subseteq \mathbb{Z}^n$ is a sublattice, its *saturation* is the lattice $\{\boldsymbol{v} \in \mathbb{Z}^n \mid \exists\, d \in \mathbb{Z},\ d \neq 0 \text{ s.t. } d\boldsymbol{v} \in L\}$.

For now we consider toric arrangements that are compatible with Lenz's definition; in Section 7.3 we will see a first generalization. Starting from a matrix $M \in \mathcal{M}_{n\times m}(\mathbb{Z})$ with columns $\boldsymbol{v}_1, \dots, \boldsymbol{v}_m$ not necessarily primitive, we define a divisorial arrangement in the torus $(\mathbb{C}^*)^n$ as

$$\mathcal{A} = \mathcal{A}_M \coloneqq \mathcal{K}_1 \cup \cdots \cup \mathcal{K}_m \tag{7.2}$$

where, for each $\boldsymbol{v}_i$, if $d_i = \mathrm{GCD}(v_{i,1}, \dots, v_{i,n})$, $\mathcal{K}_i$ is the set of layers

$$\mathcal{K}_i \coloneqq \{\mathcal{K}(\Gamma_i, \varphi_{i,j}) \mid j = 0, \dots, d_i - 1\}$$

such that $\Gamma_i = \langle \tilde{\boldsymbol{v}}_i \rangle$ and $\varphi_{i,j} \colon \tilde{\boldsymbol{v}}_i \mapsto \eta^j$, with $\tilde{\boldsymbol{v}}_i = (v_{i,1}/d_i, \dots, v_{i,n}/d_i)$ and η a primitive d_i-th root of unity. Notice that $\#(\mathcal{K}_i) = d_i$ and $\#(\mathcal{A}) = d_1 + \cdots + d_m$.

Example 7.2. Consider the matrix

$$M = \begin{pmatrix} 2 & 1 & 3 \\ 0 & 1 & 1 \end{pmatrix}.$$

The arrangement $\mathcal{A}_M$ has four layers: the first column gives rise to two of them, namely

$$\mathcal{K}_1 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 = 1\} \text{ and } \mathcal{K}_2 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 = -1\},$$

while the other two columns are primitive vectors, so they give a layer each:

$$\mathcal{K}_3 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 t_2 = 1\},$$
$$\mathcal{K}_4 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1^3 t_2 = 1\}.$$

The corresponding arrangement of $(S^1)^2$ and the poset of layers are depicted in Figure 7.2.

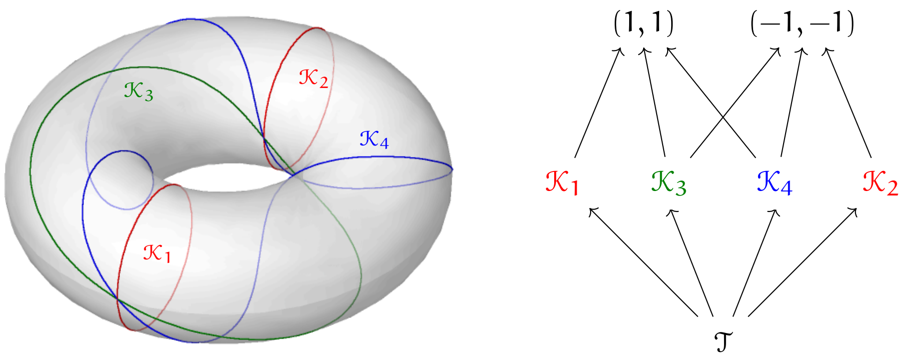


Figure 7.2: A picture of the arrangement of Example 7.2 in the compact torus $(S^1)^2$ and its poset of layers.

The next step is the computation of the variety $X_{\mathcal{A}}$, using the algorithms described in the previous chapter. As we observed above, for the arrangements $\mathcal{A}_M$ that come from a matrix $M \in \mathcal{M}_{n\times m}(\mathbb{Z})$, if $\mathcal{K}(\Gamma, \varphi) \in \mathcal{A}_M$ then $\Gamma = \langle \tilde{\boldsymbol{v}} \rangle$ where $\boldsymbol{v}$ is a column of M and $\tilde{\boldsymbol{v}} = \boldsymbol{v}/\operatorname{GCD}(\boldsymbol{v})$. So we create the fan $\Delta(\mathcal{A})$ starting from the set $\Xi = \{\tilde{\boldsymbol{v}} \mid \boldsymbol{v} \text{ is a column of } M\}$.

A presentation for the integer cohomology of a toric variety X_Δ is well-known (see for example [16, Section 12.4] or [25, Section 5.2]).

Theorem 7.2. *Let $X = X_\Delta$ be a smooth complete $\mathcal{T}$-variety. Let $\mathcal{R}$ be the set of primitive generators of the rays of Δ and define a polynomial indeterminate $C_{\mathbf{r}}$ for each $\mathbf{r} \in \mathcal{R}$. Then*

$$H^*(X_\Delta; \mathbb{Z}) \simeq \mathbb{Z}[C_{\mathbf{r}} \mid \mathbf{r} \in \mathcal{R}] / (I_{SR} + I_L)$$

where

- I_{SR} *is the* Stanley-Reisner ideal

$$I_{SR} \coloneqq (C_{\mathbf{r}_1} \cdots C_{\mathbf{r}_k} \mid \mathbf{r}_1, \ldots, \mathbf{r}_k \textit{ do not belong to a cone of } \Delta);$$

- I_L *is the* linear equivalence ideal

$$I_L \coloneqq \left(\sum_{\mathbf{r}\in\mathcal{R}} \langle \beta, \mathbf{r} \rangle C_{\mathbf{r}} \;\middle|\; \beta \in X^*(\mathcal{T}) \right).$$

Notice that for I_{SR} it is sufficient to take only the square-free monomials, and for I_L it is sufficient to take only the β's belonging to a basis of $X^*(\mathcal{T})$.

Now we have to build the wonderful model $Y_{\mathcal{A}}$. This construction actually depends on the choice of a particular subset of the poset of layers $\mathcal{C}(\mathcal{A})$.

Definition 7.3. Let Λ be a simple arrangement of subvarieties of X (see Definition 6.17). A subset $\mathcal{G} \subseteq \Lambda$ is a *building set* for Λ if for each subvariety $\Lambda_i \in \Lambda \setminus \mathcal{G}$ the minimal[▸2] elements of the set $\{G \in \mathcal{G} \mid G \supset \Lambda_i\}$ intersect transversally and their intersection is Λ_i.

Let $U \subseteq X$ be an open set. The *restriction* of an arrangement of subvarieties Λ to U is the set

$$\Lambda\big|_U \coloneqq \{\Lambda_i \cap U \mid \Lambda_i \in \Lambda,\ \Lambda_i \cap U \neq \varnothing\}.$$

Definition 7.4. Let Λ be an arrangement of subvarieties of X. A subset $\mathcal{G} \subseteq \Lambda$ is a *building set* for Λ if there is an open cover $\mathcal{U}$ of X such that

1. for every $U \in \mathcal{U}$, the restriction $\Lambda\big|_U$ is simple;

[▸2] With respect to the inclusion.

2. for every $U \in \mathcal{U}$, $\mathcal{G}|_U$ is a building set for $\Lambda|_U$.

Remark. Following [11], we consider building sets for the arrangement $\mathcal{L}'$ defined at the end of Section 6.1. If we define $\mathcal{C}_0(\mathcal{A}) \coloneqq \mathcal{C}(\mathcal{A}) \setminus \{\mathcal{T}\}$, i.e. the poset of layers of $\mathcal{A}$ without the minimum element $\mathcal{T}$, the closure operation establishes a poset isomorphism $\mathcal{L}' \simeq \mathcal{C}_0(\mathcal{A})$, therefore we consider building sets as subsets of $\mathcal{C}_0(\mathcal{A})$ itself.

In order to compute the cohomology of $Y_{\mathcal{A}}$ using the methods of [11], a building set is required to have an additional property.

Definition 7.5. A building set $\mathcal{G}$ is *well-connected* if for any subset $\{G_1, \dots, G_k\} \subseteq \mathcal{G}$, if the intersection $G_1 \cap \cdots \cap G_k$ has two or more connected components, then each of these components belongs to $\mathcal{G}$.

Example 7.3. If Λ is a simple arrangement, then each building set $\mathcal{G}$ for Λ is well-connected. In fact every intersection $G_1 \cap \cdots \cap G_k$ is either empty or connected. As none of them verifies the antecedent, the condition of Definition 7.5 is vacuously true.

Since $Y_{\mathcal{A}}$ depends on the choice of $\mathcal{G}$, we should write $Y_{\mathcal{A}}(\mathcal{G})$; however we will stay with the simpler notation $Y_{\mathcal{A}}$ and leave the building set $\mathcal{G}$ implicit when there is no ambiguity.

Remark. One may choose $\mathcal{G} = \mathcal{C}_0(\mathcal{A})$, which is always a well-connected building set. With this choice the computation $\mathcal{G}$ is avoided; however bigger building sets imply more computation, and overall more complicated wonderful model and cohomology.

The following algorithm is able to find the *minimal* well-connected building set for a given arrangement $\mathcal{A}$ defined by the matrix `M`: this allows us to compute the "simplest" wonderful model.

```
def minimal_well_connected_building_set(M):
   P=poset_of_layers(M)
   G=P.level_sets()[1] # all layers of the arrangement
   for A in Subsets(P.level_sets()[1]):
      if A.cardinality()>=2:
         Int=intersection_of_poset_subset(P,A)
         if len(Int)>=2:
            G+=list(Int)
   for level in P.level_sets()[2:]:
      for p in level:
         if p not in G:
            C=M.matrix_from_columns(list(p[0]))
            cod=C.rank()
            minimal=reduce_list([q for q in G if P.is_lequal(q,p)],lambda
             ↪  s,t: P.is_gequal(s,t))
```

```
15            ranks=sum([M.matrix_from_columns(list(q[0])).rank() for q in minimal],0)
16            if cod!=ranks:
17                G+=[p]
18    return P.subposet([P.bottom()]+G)
```

The starting point is the matrix `M` that defines the arrangement. The algorithm computes the poset of layers `P` $= \mathcal{C}(\mathcal{A})$ using an implementation of Lenz's algorithm [31] and defines a "candidate building set" `G`, in which there are the layers of $\mathcal{A}$ (`P.level_sets()[1]`).

Then the algorithm checks the "well-connected" condition (lines 4–8): for each possible intersection of layers of $\mathcal{A}$, if it has at least two connected components, the algorithm includes all of them in the candidate building set. Here we use the auxiliary function `intersection_of_poset_subset` that, given a poset `P` and a subset `A` $\subseteq$ `P`, returns the set $\{H \in \mathtt{P} \mid H \geqslant G \text{ for all } G \in \mathtt{A}\}$, i.e. in this case the set of the connected components of the intersection of all elements in `A`.

After that the algorithm decides which of the remaining elements of `P` must be included in order to have a building set. Recall that we have to check that, if `p` $\in$ `P` does *not* belong to the building set, then the minimal elements (with respect to inclusion) of the set $\mathcal{G}_\mathtt{p} \coloneqq \{G \in \mathcal{G} \mid G \supseteq \mathtt{p}\}$ intersect transversally and their intersection is `p` (see Definition 7.3). Notice that to check this condition it suffices to consider only the elements $G \in \mathcal{G}$ such that $\mathrm{rk}(G) < \mathrm{rk}(\mathtt{p})$.[3] Therefore the algorithm produces the candidate building set one level at a time, starting from rank two (line 9). For each element `p` not belonging to the candidate building set, the algorithm computes its codimension (lines 12–13) and the minimal elements of the set $\mathcal{G}_\mathtt{p}$ (line 14). In this line, `reduce_list` is a generalization of `reduce_ideal_list` of Section 4.2: it takes a list L of elements and a boolean-valued function f with two inputs of L and returns a new list $L' = [\ell \in L \mid \nexists m \in L \text{ s.t. } f(\ell, m) = \mathtt{True}]$. For this algorithm we use the function defined as $f(G_1, G_2) = \mathtt{True}$ if and only if $G_1 \geqslant G_2$ in $\mathcal{C}(\mathcal{A})$, that is to say $G_1 \subseteq G_2$.

The intersection of the elements in `minimal` is guaranteed to be *exactly* `p`, because otherwise it would be disconnected and `p` would be put in `G` during the cycle of lines 4–8. The only failure happens if this intersection is not transversal, i.e. the sum of codimensions of the elements in `minimal` is different than the codimension of `p`. In this case, `p` must be added to the elements of `G` (lines 15–17); otherwise, we may choose to add it or not. Actually the algorithm always chooses *not* to add it, thus computing the minimal building set: it is possible to modify the algorithm, adding an `else` condition between lines 17 and 18, in order to let the user decide.

Finally the algorithm returns the subposet generated by `G` and the whole torus (`P.bottom()`). Recall that it does *not* actually belong to the building set, but we include

[3] This is the rank of $\mathcal{C}(\mathcal{A})$: remember that it is a poset ranked by codimension.

it here because of how subsequent algorithms do their computations.

We are ready to show the presentation of $H^*(Y_{\mathcal{A}};\mathbb{Z})$ obtained by De Concini and Gaiffi in [11]. Let $X_{\mathcal{A}}$ be a good toric variety for $\mathcal{A}$ and let $B \coloneqq H^*(X_{\mathcal{A}};\mathbb{Z})$ be its cohomology ring. For each $G \in \mathcal{G}$ let T_G be a polynomial indeterminate. We are going to produce an ideal I_W of the polynomial ring $B[T_G \mid G \in \mathcal{G}]$ such that the cohomology ring of $Y_{\mathcal{A}}$ is isomorphic to the quotient ${}^{B[T_G \mid G \in \mathcal{G}]}/_{I_W}$. To do so, we need some auxiliary polynomials.

Let Z be an indeterminate and, for every $G \in \mathcal{C}(\mathcal{A})$ denote by Γ_G the lattice such that $G = \mathcal{K}(\Gamma_G, \varphi)$. For every pair $(M, G) \in \mathcal{C}(\mathcal{A}) \times \mathcal{C}(\mathcal{A})$ with $M \leqslant G$ (i.e. $G \subseteq M$) choose a basis $(\beta_1, \dots, \beta_s)$ for Γ_G such that $(\beta_1, \dots, \beta_k)$, with $k \leqslant s$, is a basis for Γ_M. If M is the whole torus $\mathcal{T}$, then choose any basis of Γ_G and let $k = 0$. Define the polynomials $P_G^M \in B[Z]$ as

$$P_G^M \coloneqq \prod_{j=k+1}^{s} \left(Z - \sum_{\mathbf{r}\in\mathcal{R}} \min(0, \langle \beta_j, \mathbf{r} \rangle) C_{\mathbf{r}} \right). \tag{7.3}$$

Notice that we allow $G = M$: in that case $P_G^G \coloneqq 1$ since it is an empty product.

Now consider the following set:[4]

$$\mathcal{W} \coloneqq \left\{ (G, A) \in \mathcal{G} \times \wp(\mathcal{G}) \;\middle|\; G \subsetneq \mathcal{K} \text{ for all } \mathcal{K} \in A \right\}.$$

For each $G \in \mathcal{G}$ define

$$B_G \coloneqq \{H \in \mathcal{G} \mid H \subseteq G\} \tag{7.4}$$

and for each $(G, A) \in \mathcal{W}$ with $A = \{G_1, \dots, G_k\}$ let M be the unique connected component of $G_1 \cap \dots \cap G_k$ that contains G (if $A = \varnothing$, let $M = \mathcal{T}$). Define the polynomial in $B[T_G \mid G \in \mathcal{G}]$

$$F(G, A) \coloneqq P_G^M \Big(\sum_{H \in B_G} -T_H \Big) \prod_{K \in A} T_K. \tag{7.5}$$

Finally let $\mathcal{W}_0 \coloneqq \{A = \{G_1, \dots, G_k\} \in \wp(\mathcal{G}) \mid G_1 \cap \dots \cap G_k = \varnothing\}$. For each $A \in \mathcal{W}_0$ define the polynomial in $B[T_G \mid G \in \mathcal{G}]$

$$F(A) \coloneqq \prod_{K \in A} T_K. \tag{7.6}$$

Theorem 7.6 ([11, Theorem 7.1]). *The cohomology ring* $H^*(Y_{\mathcal{A}};\mathbb{Z})$ *is isomorphic to the quotient of* $B[T_G \mid G \in \mathcal{G}]$ *by the ideal* I_W *generated by*

1. *the products* $C_{\mathbf{r}} T_G$, *with* $G \in \mathcal{G}$ *and* $\mathbf{r} \in \mathcal{R}$ *such that* $\mathbf{r}$ *does* not *belong to* V_{Γ_G};[5]

[4] For a set $\mathcal{X}$, we denote by $\wp(\mathcal{X})$ the power set of $\mathcal{X}$.
[5] Recall that $V_\Gamma \coloneqq \{\mathbf{v} \in V \mid \langle \chi, \mathbf{v} \rangle = 0 \text{ for all } \chi \in \Gamma\}$.

2. *the polynomials* $F(G, A)$ *for every pair* $(G, A) \in \mathcal{W}$;

3. *the polynomials* $F(A)$ *for every* $A \in \mathcal{W}_0$.

Putting all together, we have

$$H^*(Y_{\mathcal{A}}; \mathbb{Z}) \simeq {}^{B[T_G \mid G \in \mathcal{G}]}\big/_{I_W} \simeq {}^{\mathbb{Z}[C_r, T_G \mid r \in \mathcal{R}, G \in \mathcal{G}]}\big/_{(I_{SR} + I_L + I_W)}.$$

Remark. It is known (see [12, Theorem 9.1]) that the cohomology of the projective wonderful model $Y_{\mathcal{A}}$ satisfies:

- $H^i(Y_{\mathcal{A}}; \mathbb{Z}) = 0$ for i odd;
- $H^i(Y_{\mathcal{A}}; \mathbb{Z})$ is torsion-free for i even.

Notice that the ideal I_W is homogeneous and that the image of an indeterminate T_G under the isomorphism stated in Theorem 7.6 belongs to $H^2(Y_{\mathcal{A}}; \mathbb{Z})$ ([11, Theorem 7.1] describes this isomorphism explicitly). This means that $H^*(Y_{\mathcal{A}}; \mathbb{Z})$ and ${}^{B[T_G \mid G \in \mathcal{G}]}\big/_{I_W}$ are isomorphic as *graded* rings and that

$$\left({}^{B[T_G \mid G \in \mathcal{G}]}\big/_{I_W}\right)_i \simeq H^{2i}(Y_{\mathcal{A}}; \mathbb{Z}).$$

In particular, if $\mathcal{B}$ is a monomial basis of ${}^{B[T_G \mid G \in \mathcal{G}]}\big/_{I_W}$ as $\mathbb{Z}$-module, we have

$$\mathrm{rk}(H^{2i}(Y_{\mathcal{A}}; \mathbb{Z})) = \#(\{m \in \mathcal{B} \mid \deg(m) = i\}).$$

We have implemented in the `SageMath` language all the previous steps. The following code takes as input an integer matrix `Mat` $\in \mathcal{M}_{n \times m}(\mathbb{Z})$ representing a toric arrangement in $(\mathbb{C}^*)^n$ and returns the ideal $I_{SR} + I_L + I_W$. It takes also a boolean flag `minimal_building` that controls the building set: if the flag is `True`, the algorithm computes the minimal one using `minimal_well_connected_building_set`; otherwise, it uses the whole $\mathcal{C}_0(\mathcal{A})$.

```
def wonderful_cohomology_ring(Mat,minimal_building):
    P=poset_of_layers(Mat)
    if minimal_building:
        G=minimal_well_connected_building_set(Mat)
    else:
        G=copy(P)
    primitive_vectors=[vector([ci//GCD(C) for ci in C]) for C in
     ↪ Mat.columns()]
    n=Mat.nrows() # dimension of torus
    if n==2: # choose algorithm
        F=smooth_equal_sign_fan(primitive_vectors)
```

```
    else:
        P1n=build_fan_P1n(n)
        F=dcg_algorithm(P1n,primitive_vectors)
    bases_dict=bases_dictionary(Mat)
    r=F.nrays()
    g=G.cardinality()-1
    R=PolyRingCohomology(r,g) # Q[c1,...,cr,t1,...,tg]
    c=[1]+list(R.gens()[:r]) # so that c[i]=c_i
    t=[1]+list(R.gens()[r:]) # so that t[i]=t_i
    poly_dict={}
    RR=PolynomialRing(R,"z")
    z=RR.gen()
    vec_rays=[vector(tuple(v)) for v in F.rays()]
    for pair in bases_dict.iterkeys():
        poly_dict[pair]=prod([z-sum([min(0,vec.dot_product(vec_rays[i]))*c[i+1] for i in xrange(r)],0) for vec in bases_dict[pair][1]],1)
    for p in P:
        poly_dict[(p,p)]=RR.one()
    RC=PolynomialRing(QQ,["x_{}".format(i+1) for i in xrange(r)]) # Q[x_1,...,x_r]
    ISR=[rel(c[1:]) for rel in F.Stanley_Reisner_ideal(RC).gens()]
    IL=[rel(c[1:]) for rel in F.linear_equivalence_ideal(RC).gens()]
    ordered_G=sum(G.level_sets()[1:],[])
    CT=[]
    for p in ordered_G:
        MM=[primitive_vectors[v] for v in p[0]]
        for ray in vec_rays:
            found=False
            for vec in MM:
                if vec.dot_product(ray)!=0:
                    found=True
                    break
            if found:
                CT+=[c[vec_rays.index(ray)+1]*t[ordered_G.index(p)+1]]
    FF=[]
    for p in ordered_G:
        Asets=Subsets([q for q in G.principal_order_ideal(p) if (q!=p and q!=G[0])])
        Bi=[q for q in G.principal_order_filter(p) if q!=G[0]]
```

```
        for A in Asets:
            if len(A)==0:
                M=G[0]
            else:
                Aints=[q for q in intersection_of_poset_subset(P,A) if q!=P[0]] # note the poset P here
                for q in Aints:
                    if P.is_lequal(q,p):
                        M=q
                        break
            poly=poly_dict[(M,p)]
            FF+=[poly(sum([-t[ordered_G.index(h)+1] for h in Bi],0))*prod([t[ordered_G.index(j)+1] for j in A],1)]
    FA=[]
    for A in Subsets(ordered_G):
        try:
            Aints=[q for q in intersection_of_poset_subset(P,A) if q!=P[0]]
        except ValueError: continue
        if len(Aints)==0:
            FA+=[prod([t[ordered_G.index(j)+1] for j in A],1)]
    return R.ideal(ISR+IL+CT+FF+FA)
```

The algorithm `wonderful_cohomology_ring` is quite straightforward. First of all it computes $P = \mathcal{C}(\mathcal{A})$ (line 2) and chooses the building set $G = \mathcal{G}$, depending on the value of `minimal_building` (lines 3–6). Then it computes the fan associated with a good toric variety for $\mathcal{A}$ (lines 7–13): it creates the list `primitive_vectors` that contains the columns of `Mat`, each divided by the GCD of its elements. Depending on the dimension of the torus, the algorithm chooses either `smooth_equal_sign_fan` or `dcg_algorithm` described in the previous chapter. In line 12, the auxiliary algorithm `build_fan_P1n` returns the fan Ω generated by the orthants of $\mathbb{R}^n$, which is used to start `dcg_algorithm`.

Line 14 uses the algorithm `bases_dictionary`, which returns a (Python) dictionary `bases_dict` indexed by pairs $(M, G) \in \mathcal{C}(\mathcal{A}) \times \mathcal{C}(\mathcal{A})$ with $M < G$ such that `bases_dict[`(M, G)`]` is a pair of lists of vectors $([v_1, \dots, v_k], [v_{k+1}, \dots, v_s])$ with $\Gamma_M = \langle v_1, \dots, v_k \rangle$ and $\Gamma_G = \langle v_1, \dots, v_s \rangle$. This algorithm is important *per se*, so we report its code here.

```
def bases_dictionary(Mat):
    n=Mat.nrows()
    P=poset_of_layers(Mat)
    layer_basis={}
    for p in P:
```

```
6          Mp=Mat.matrix_from_columns(list(p[0]))
7          HF=Mp.transpose().hermite_form()
8          layer_basis[p]=matrix([[ri//GCD(row) for ri in row] for row in HF.rows()]).transpose() # the COLUMNS are a basis
9      bases_dict={}
10     for p in P:
11         for q in P:
12             if p==P[0] and q!=P[0]:
13                 bases_dict[(p,q)]=([],[vector(c) for c in layer_basis[q].columns()])
14             elif P.is_less_than(p,q):
15                 A=layer_basis[p]
16                 k=A.ncols()
17                 B=layer_basis[q]
18                 s=B.ncols()
19                 R=A.smith_form()[1] # matrix of the row operations
20                 C=zero_matrix(k,s).stack((R*B).matrix_from_rows(range(k,n)))
21                 HF=C.transpose().hermite_form(include_zero_rows=False).transpose()
22                 newC=R.inverse()*HF
23                 bases_dict[(p,q)]=([vector(c) for c in A.columns()],[vector(c) for c in newC.columns()])
24     return bases_dict
```

After the computation of the poset of layers `P`, this algorithm creates a dictionary `layer_basis` indexed by the elements of $\mathcal{C}(\mathcal{A})$ such that, for $G \in \mathcal{C}(\mathcal{A})$, `layer_basis[G]` is a matrix whose columns form a basis for Γ_G (lines 4–8). To do so, it takes from `Mat` the columns corresponding to the layers of $\mathcal{A}_{\texttt{Mat}}$ that give G (line 6)[▸6] and computes the Hermite form.[▸7] The rows of this matrix are not necessarily primitive: in fact, when the algorithm extracts the generators of Γ_G in line 4, it takes them from the original matrix `Mat`, thus in some sense it considers *all* the connected components of the intersections of the hypersurfaces S_i (defined in Equation (7.1)), among which there is G. More precisely, suppose that $\texttt{p[0]} = \{i_1, \dots, i_k\}$: then the corresponding columns of `Mat` define all the connected components of $S_{i_1} \cap \dots \cap S_{i_k}$. This is not a problem, because all these layers share the same lattice Γ_G, which can be found by "primitivizing" the

[▸6] If `P` is the output of `poset_of_layers(Mat)`, then an element $\texttt{p} \in \texttt{P}$ is a pair such that the first component contains the indices of the columns of `Mat` corresponding to the 1-codimensional layers whose intersection has `p` as a connected component.

[▸7] In `SageMath`, the method `hermite_form` does *row* operations, so we have to transpose the matrix before computing it.

generators. Therefore the algorithm divides each row of the Hermite form by the GCD of its coefficients (line 8), recovering a basis of the lattice Γ_G.

Once that we have bases for all the layers in the poset, the algorithm defines a new dictionary `bases_dict` and goes through each pair $(M, G) \in \mathcal{C}(\mathcal{A}) \times \mathcal{C}(\mathcal{A})$ (in the algorithm, they correspond to the pairs $(\mathtt{p}, \mathtt{q}) \in \mathtt{P} \times \mathtt{P}$). For the pairs $(\mathcal{T}, G)$ the algorithm defines `bases_dict[`$(\mathcal{T}, G)$`]` $=$ [[ ], [columns of `layer_basis[G]`]]; for the pairs (M, G) with $M < G$, it implements the following procedure.

If $M < G$, then $\Gamma_M \subseteq \Gamma_G$, so $\Gamma_M + \Gamma_G = \Gamma_G$. Suppose that $(\beta_1, \dots, \beta_k)$ is a basis for Γ_M and $(\gamma_1, \dots, \gamma_s)$ is a basis for Γ_G, where the β_i's and the γ_j's are the columns of `layer_basis[M]` and `layer_basis[G]` respectively (in particular $k < s$). It follows that $\{\beta_1, \dots, \beta_k, \gamma_1, \dots, \gamma_s\}$ is a set of generators for Γ_G. Consider the matrices

$$A = \begin{pmatrix} | & & | \\ \beta_1 & \cdots & \beta_k \\ | & & | \end{pmatrix} \in \mathcal{M}_{n \times k}(\mathbb{Z}), \qquad B = \begin{pmatrix} | & & | \\ \gamma_1 & \cdots & \gamma_s \\ | & & | \end{pmatrix} \in \mathcal{M}_{n \times s}(\mathbb{Z})$$

and put them aside to form the matrix

$$[A|B] = \left(\begin{array}{ccc|ccc} | & & | & | & & | \\ \beta_1 & \cdots & \beta_k & \gamma_1 & \cdots & \gamma_s \\ | & & | & | & & | \end{array}\right) \in \mathcal{M}_{n \times (k+s)}(\mathbb{Z}).$$

Change basis so that Γ_M becomes the lattice generated by the first k vectors of the standard basis of $\mathbb{Z}^n$. In order to do so, compute the Smith form of the matrix A. In particular, find two unimodular matrices $R \in \mathcal{M}_{n \times n}(\mathbb{Z})$, $U \in \mathcal{M}_{k \times k}(\mathbb{Z})$ such that

$$R\,A\,U = \begin{pmatrix} I_k \\ 0 \end{pmatrix}$$

where I_k is the $k \times k$ identity matrix and 0 is a zero $(n-k) \times k$ block. This change of bases applied to the whole matrix $[A|B]$ becomes

$$R\,[A|B] \begin{pmatrix} U & 0 \\ 0 & I_s \end{pmatrix} = \left(\begin{array}{c|c} I_k & \multirow{2}{*}{RB} \\ 0 & \end{array}\right).$$

Use the first k columns of this matrix to kill the upper part of the matrix RB, obtaining a matrix of the form

$$\begin{pmatrix} 0 \\ C \end{pmatrix}$$

with a zero $k \times s$ block and $C \in \mathcal{M}_{(n-k) \times s}(\mathbb{Z})$. After all these operations, we have that the columns of the matrix

$$\begin{pmatrix} I_k & 0 \\ 0 & C \end{pmatrix}$$

are a set of generators for Γ_G: to recover a basis, compute the Hermite form *with column operations*, obtaining a new matrix

$$\begin{pmatrix} I_k & 0 \\ 0 & \tilde{C} \end{pmatrix}$$

where *only* $s-k$ *columns of* $\tilde{C}$ *are different than zero* (because the rank of Γ_G is s). Denote by C' the $(n-k)\times(s-k)$ matrix obtained from $\tilde{C}$ by removing the zero columns; the columns of the $n\times s$ matrix

$$R^{-1}\begin{pmatrix} I_k & 0 \\ 0 & C' \end{pmatrix}\begin{pmatrix} U^{-1} & 0 \\ 0 & I_{(s-k)} \end{pmatrix} = \left(A \,\middle|\, R^{-1}\begin{pmatrix} 0 \\ C' \end{pmatrix} \right)$$

are a basis of Γ_G, and the first k columns are a basis of Γ_M.

Now we come back to the description of `wonderful_cohomology_ring`. The algorithm collects all the information it needs to compute the polynomial ring $\mathbb{Z}[C_{\mathbf{r}}, T_G \mid \mathbf{r} \in \mathcal{R}, G \in \mathcal{G}]$: the number of rays of the fan `r` (line 15) and the number of elements in the building set $\mathcal{G}$, which is $\mathtt{g} = \#(\mathcal{G}) - 1$ (line 16). `PolyRingCohomology(r,g)` is an auxiliary function that takes two integers `r` and `g` and returns the ring

$$\mathtt{R} = \mathbb{Q}[\mathtt{c}_1, \ldots, \mathtt{c}_\mathtt{r}, \mathtt{t}_1, \ldots, \mathtt{t}_\mathtt{g}].$$

The indeterminates are stored in two lists for later use.

The algorithm then proceeds to the computation of the polynomials P_G^M (lines 21–27). This is done by defining another dictionary `poly_dict`, indexed by the pairs $(M, G) \in \mathcal{C}(\mathcal{A}) \times \mathcal{C}(\mathcal{A})$ such that $M \leqslant G$, where $\mathtt{poly\_dict[(M,G)]} = P_G^M$. This is quite easy: after the ring definition $\mathtt{RR} = \mathtt{R}[z]$ (it includes also the variables $\mathtt{t}_1, \ldots, \mathtt{t}_\mathtt{g}$, even if they are not used for the P_G^M's) and a technical passage to convert the rays to the type `vector`, the algorithm populates the dictionary using the definition of the P_G^M's (Equation (7.3)). Note that the polynomials $P_G^G = 1$ are included (lines 26–27).

Once that the setup is complete, the algorithm begins to compute the ideal $(I_{SR} + I_L + I_W)$. The first two ideals are easy, because `SageMath` has already the methods `Stanley_Reisner_ideal` and `linear_equivalence_ideal` that compute I_{SR} and I_L respectively. In order to keep things clean, we define a new polynomial ring $\mathtt{RC} = \mathbb{Q}[\mathtt{x}_1, \ldots, \mathtt{x}_\mathtt{r}]$ and compute the relations in this ring, evaluating $\mathtt{x}_i = \mathtt{c}_i$ only in a second moment.

The only computation left is I_W. There are three types of relations, and the algorithm computes them one type at a time. First of all, we fix an auxiliary list `ordered_G` which contains the elements of $\mathcal{G}$ (without $\mathcal{T}$) ordered in such a way that if $\mathtt{ordered\_G}[i] < \mathtt{ordered\_G}[j]$ then $i < j$ (line 31). This helps the user of this algorithm to identify the element $\mathtt{p} \in \mathcal{G}$ corresponding to a variable $\mathtt{t}_i$: it is $\mathtt{ordered\_G}[i-1]$ (recall that `SageMath` indexes the lists starting from 0).

1. The products $C_{\mathbf{r}}T_G$ are computed in lines 32–42. For each $G \in \mathcal{G}$ the algorithm chooses a set of generators `MM` and begins to test each ray. The product $C_{\mathbf{r}}T_G$ is included only if it does not belong to V_{Γ_i}, i.e. there is a vector `vec` in `MM` such that $\langle \texttt{vec}, \mathbf{r} \rangle \neq 0$.

2. The polynomials $F(G, A)$ for $(G, A) \in \mathcal{W}$ are computed in lines 43–57. For each $G \in \mathcal{G}$, `Asets` is the set $\{A \in \wp(\mathcal{G}) \mid (G, A) \in \mathcal{W}\}$ and `Bi` is the set B_G defined in (7.4). Then, for each $A \in$ `Asets`, the algorithm computes the corresponding M (note the use of `P` in lines 51 and 53: this is because the intersection of the elements of A needn't belong to $\mathcal{G}$), recovers the polynomial `poly` $= P_G^M$ and adds $F(G, A)$ to the relations in `FF`.

3. The polynomials $F(A)$ for $A \in \mathcal{W}_0$ are computed in lines 58–64. The function `intersection_of_poset_subset` raises a `ValueError` if `A` is empty, therefore in that case the algorithm continues with the next subset; if the intersection of the elements of `A` is empty, the corresponding polynomial $F(A)$ is added to the relations in `FA`.

Finally the algorithm returns the ideal generated by all the relations found.

7.2 Examples

In this section we show some small examples of cohomology rings of projective wonderful models. In all of them, except the last one, we use the whole poset $\mathcal{C}_0(\mathcal{A})$ as the well-connected building set for the wonderful model $Y_{\mathcal{A}}$.

Example 7.4. For the first example, which we are going to analyse in detail, we use again the arrangement of Example 7.1; recall that it has two layers

$$\mathcal{K}_1 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 = 1\} \text{ and } \mathcal{K}_2 = \{(t_1, t_2) \in (\mathbb{C}^*)^2 \mid t_1 t_2^2 = 1\}$$

that intersect in the two points $P_1 = (1, 1)$ and $P_2 = (1, -1)$. In particular, the arrangement $\mathcal{A}$ is defined by the matrix

$$\begin{pmatrix} 1 & 1 \\ 0 & 2 \end{pmatrix}.$$

Now we follow the algorithm. First of all, it computes the poset of layers $\mathcal{C}(\mathcal{A})$, which is shown in Figure 7.3 in the middle. After that we tell the algorithm to choose the building set $\mathcal{G} = \mathcal{C}_0(\mathcal{A})$, but in this small example that makes no difference: $\mathcal{C}_0(\mathcal{A})$ is the only well-connected building set. For, let $\mathcal{G}$ be any such set: the two layers $\mathcal{K}_1$ and $\mathcal{K}_2$ belong to $\mathcal{G}$ by default, and their intersection is disconnected, so both P_1 and P_2 must belong to $\mathcal{G}$.

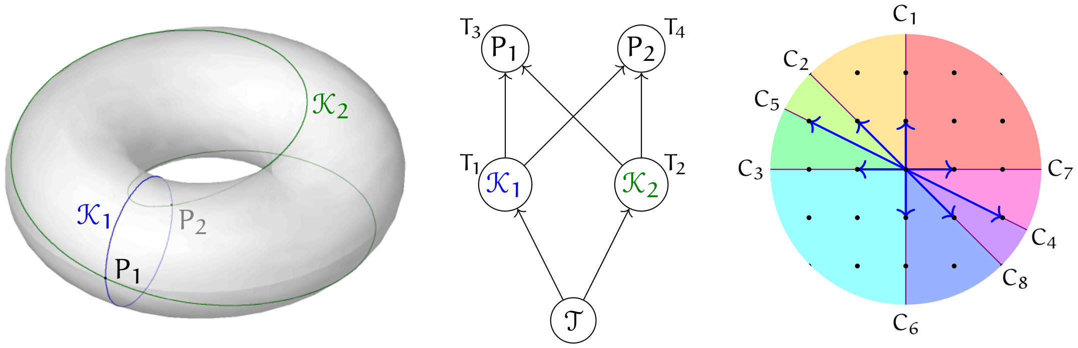


Figure 7.3: Real picture in $(S^1)^2$ of the arrangement $\mathcal{A}$ associated with the vectors $(1,0)$ and $(1,2)$, together with the poset of layers $\mathcal{C}(\mathcal{A})$ and the fan for $X_{\mathcal{A}}$.

Then the algorithm computes the list of generators of the lattices associated with the two layers $\mathcal{K}_1$ and $\mathcal{K}_2$. In this case the columns of the matrix are primitive, so `primitive_vectors` $= [(1,0),(1,2)]$.

The next step is the computation of the fan. The arrangement lives in a 2-dimensional torus, so the algorithm computes the fan using `smooth_equal_sign_fan`. The result can be seen on the right in Figure 7.3.

After that we have to compute the bases dictionary. There are eight pairs $(M,G) \in \mathcal{C}(\mathcal{A}) \times \mathcal{C}(\mathcal{A})$ with $M < G$, and we report here directly the result of `bases_dictionary`:

$$
\begin{aligned}
(\mathcal{T},\mathcal{K}_1)&\colon [\varnothing,[(1,0)]], & (\mathcal{T},\mathcal{K}_2)&\colon [\varnothing,[(1,2)]],\\
(\mathcal{T},P_1)&\colon [\varnothing,[(1,0),(0,1)]], & (\mathcal{T},P_2)&\colon [\varnothing,[(1,0),(0,1)]],\\
(\mathcal{K}_1,P_1)&\colon [[(1,0)],[(0,1)]], & (\mathcal{K}_1,P_2)&\colon [[(1,0)],[(0,1)]],\\
(\mathcal{K}_2,P_1)&\colon [[(1,2)],[(0,1)]], & (\mathcal{K}_2,P_2)&\colon [[(1,2)],[(0,1)]].
\end{aligned}
$$

Now the algorithm defines the polynomial ring. There are eight variables $C_1,\dots,C_8$ associated with the rays of the fan and four variables $T_1,\dots,T_4$ associated with the building set $\mathcal{G} = \mathcal{C}(\mathcal{A}) \setminus \{\mathcal{T}\}$. For this example, the association is defined in Figure 7.3, where the variables are written next to the elements they refer to.

It is time to define the polynomials P_G^M. If $M = G$, the polynomial is set to 1; the other cases are

$$
\begin{aligned}
P_{\mathcal{K}_1}^{\mathcal{T}} &= Z + C_2 + C_3 + 2C_5,\\
P_{\mathcal{K}_2}^{\mathcal{T}} &= Z + C_3 + 2C_6 + C_8,\\
P_{P_1}^{\mathcal{T}} &= (Z + C_4 + C_6 + C_8)(Z + C_2 + C_3 + 2C_5),\\
P_{P_2}^{\mathcal{T}} &= (Z + C_4 + C_6 + C_8)(Z + C_2 + C_3 + 2C_5),\\
P_{P_1}^{\mathcal{K}_1} &= Z + C_4 + C_6 + C_8,\\
P_{P_2}^{\mathcal{K}_1} &= Z + C_4 + C_6 + C_8,
\end{aligned}
$$

$$P^{\mathcal{K}_2}_{P_1} = Z + C_4 + C_6 + C_8,$$
$$P^{\mathcal{K}_2}_{P_2} = Z + C_4 + C_6 + C_8.$$

Once that all ingredients have been set up, the algorithm begins to compute the relations. At first it calls `Stanley_Reisner_ideal` and `linear_equivalence_ideal` to compute I_{SR} and I_L respectively:

$$\begin{aligned} I_{SR} = (&C_1C_3,\ C_1C_4,\ C_1C_5,\ C_1C_6,\ C_1C_8,\ C_2C_3,\ C_2C_4,\ C_2C_6,\ C_2C_7,\ C_2C_8,\ C_3C_4,\\ &C_3C_7,\ C_3C_8,\ C_4C_5,\ C_4C_6,\ C_5C_6,\ C_5C_7,\ C_5C_8,\ C_6C_7,\ C_7C_8),\\ I_L = (&-C_2 - C_3 + 2C_4 - 2C_5 + C_7 + C_8,\ C_1 + C_2 - C_4 + C_5 - C_6 - C_8). \end{aligned}$$

Then the algorithm computes the products $C_{\mathbf{r}}T_G$ with $G \in \mathcal{G}$ and $\mathbf{r} \in \mathcal{R}$ such that $\mathbf{r}$ does *not* belong to V_{Γ_G}. In this example,

1. $V_{\Gamma_{\mathcal{K}_1}}$ is the $\mathbb{R}$-span of $(0,1)$, so only two rays do belong to it: $(0,1)$ and $(0,-1)$, corresponding to the variables C_1 and C_6;
2. $V_{\Gamma_{\mathcal{K}_2}}$ is the $\mathbb{R}$-span of $(-2,1)$, so only two rays do belong to it: $(-2,1)$ and $(2,-1)$, corresponding to the variables C_5 and C_4;
3. $V_{\Gamma_{P_1}}$ and $V_{\Gamma_{P_2}}$ are the 0-dimensional vector space $\{(0,0)\}$, so no ray belongs to them.

It follows that the relations of the form $C_{\mathbf{r}}T_G$ are

$$\begin{gathered} C_2T_1,\ C_3T_1,\ C_4T_1,\ C_5T_1,\ C_7T_1,\ C_8T_1,\\ C_1T_2,\ C_2T_2,\ C_3T_2,\ C_6T_2,\ C_7T_2,\ C_8T_2,\\ C_1T_3,\ C_2T_3,\ C_3T_3,\ C_4T_3,\ C_5T_3,\ C_6T_3,\ C_7T_3,\ C_8T_3,\\ C_1T_4,\ C_2T_4,\ C_3T_4,\ C_4T_4,\ C_5T_4,\ C_6T_4,\ C_7T_4,\ C_8T_4. \end{gathered} \tag{7.7}$$

After that there is the computation of the relations $F(G,A)$. In this example,

$$\begin{aligned} \mathcal{W} = \{&(\mathcal{K}_1,\varnothing),(\mathcal{K}_2,\varnothing),(P_1,\varnothing),(P_1,\{\mathcal{K}_1\}),(P_1,\{\mathcal{K}_2\}),(P_1,\{\mathcal{K}_1,\mathcal{K}_2\}),\\ &(P_2,\varnothing),(P_2,\{\mathcal{K}_1\}),(P_2,\{\mathcal{K}_2\}),(P_2,\{\mathcal{K}_1,\mathcal{K}_2\})\}. \end{aligned}$$

- For $G = \mathcal{K}_1$, we have $B_{\mathcal{K}_1} = \{\mathcal{K}_1, P_1, P_2\}$. The only pair of $\mathcal{W}$ with $G = \mathcal{K}_1$ has $A = \varnothing$, therefore the algorithm computes

$$F(\mathcal{K}_1,\varnothing) = P^{\mathcal{T}}_{\mathcal{K}_1}(-T_1 - T_3 - T_4) = C_2 + C_3 + 2C_5 - T_1 - T_3 - T_4.$$

- For $G = \mathcal{K}_2$, we have $B_{\mathcal{K}_2} = \{\mathcal{K}_2, P_1, P_2\}$. The only pair of $\mathcal{W}$ with $G = \mathcal{K}_2$ has $A = \varnothing$, therefore the algorithm computes

$$F(\mathcal{K}_2,\varnothing) = P^{\mathcal{T}}_{\mathcal{K}_2}(-T_2 - T_3 - T_4) = C_3 + 2C_6 + C_8 - T_2 - T_3 - T_4.$$

- For $G = P_1$, we have $B_{P_1} = \{P_1\}$ and four pairs in $\mathcal{W}$ with $G = P_1$:
  - $A = \varnothing$: the algorithm sets $M = \mathcal{T}$ and computes
  $$F(P_1, \varnothing) = P_{P_1}^{\mathcal{T}}(-T_3) = (C_4 + C_6 + C_8 - T_3)(C_2 + C_3 + 2C_5 - T_3);$$
  - $A = \{\mathcal{K}_1\}$: the algorithm sets $M = \mathcal{K}_1$ and computes
  $$F(P_1, \{\mathcal{K}_1\}) = P_{P_1}^{\mathcal{K}_1}(-T_3)\, T_1 = T_1(C_4 + C_6 + C_8 - T_3);$$
  - $A = \{\mathcal{K}_2\}$: the algorithm sets $M = \mathcal{K}_2$ and computes
  $$F(P_1, \{\mathcal{K}_2\}) = P_{P_1}^{\mathcal{K}_2}(-T_3)\, T_2 = T_2(C_4 + C_6 + C_8 - T_3);$$
  - $A = \{\mathcal{K}_1, \mathcal{K}_2\}$: the algorithm sets $M = P_1$ and computes
  $$F(P_1, \{\mathcal{K}_1, \mathcal{K}_2\}) = P_{P_1}^{P_1}(-T_3)\, T_1 T_2 = T_1 T_2.$$
- For $G = P_2$, we have $B_{P_2} = \{P_2\}$ and four pairs in $\mathcal{W}$ with $G = P_2$:
  - $A = \varnothing$: the algorithm sets $M = \mathcal{T}$ and computes
  $$F(P_2, \varnothing) = P_{P_2}^{\mathcal{T}}(-T_4) = (C_4 + C_6 + C_8 - T_4)(C_2 + C_3 + 2C_5 - T_4);$$
  - $A = \{\mathcal{K}_1\}$: the algorithm sets $M = \mathcal{K}_1$ and computes
  $$F(P_2, \{\mathcal{K}_1\}) = P_{P_2}^{\mathcal{K}_1}(-T_4)\, T_1 = T_1(C_4 + C_6 + C_8 - T_4);$$
  - $A = \{\mathcal{K}_2\}$: the algorithm sets $M = \mathcal{K}_2$ and computes
  $$F(P_2, \{\mathcal{K}_2\}) = P_{P_2}^{\mathcal{K}_2}(-T_4)\, T_2 = T_2(C_4 + C_6 + C_8 - T_4);$$
  - $A = \{\mathcal{K}_1, \mathcal{K}_2\}$: the algorithm sets $M = P_2$ and computes
  $$F(P_2, \{\mathcal{K}_1, \mathcal{K}_2\}) = P_{P_2}^{P_2}(-T_4)\, T_1 T_2 = T_1 T_2.$$

Finally there are the relations $F(A)$ for $A \in \mathcal{W}_0$. The elements of $\mathcal{C}_0(\mathcal{A})$ that don't intersect are the ones involving both P_1 and P_2, so that

$$\mathcal{W}_0 = \{\{P_1, P_2\}, \{\mathcal{K}_1, P_1, P_2\}, \{\mathcal{K}_2, P_1, P_2\}, \{\mathcal{K}_1, \mathcal{K}_2, P_1, P_2\}\}.$$

Therefore the algorithm adds the relations

$$T_3 T_4,\ T_1 T_3 T_4,\ T_2 T_3 T_4,\ T_1 T_2 T_3 T_4. \tag{7.8}$$

The last step is the production of the ideal of all the relations: the algorithm sums I_{SR}, I_L, the products of (7.7), the ten relations of the form $F(G, A)$ and the relations (7.8) of type $F(A)$ and returns the ideal generated by all these. After a computation of a Gröbner basis DEGREVLEX[8] we conclude that

$$H^*(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}[C_1,\dots,C_8,T_1,\dots,T_4]/_I$$

where

$$\begin{aligned} I = (&T_4^3,\ C_5^2 - T_4^2,\ C_5C_6,\ C_6^2 - T_4^2,\ C_5C_7,\ C_6C_7,\ C_7^2 - 2T_4^2,\ C_5C_8,\ C_6C_8 + T_4^2,\\ &C_7C_8,\ C_8^2 - 2T_4^2,\ C_5T_1,\ C_6T_1 + T_4^2,\ C_7T_1,\ C_8T_1,\ T_1^2 - 2T_4^2,\ C_5T_2 + T_4^2,\\ &C_6T_2,\ C_7T_2,\ C_8T_2,\ T_1T_2,\ T_2^2 - 2T_4^2,\ C_5T_3,\ C_6T_3,\ C_7T_3,\ C_8T_3,\ T_1T_3 + T_4^2,\\ &T_2T_3 + T_4^2,\ T_3^2 - T_4^2,\ C_5T_4,\ C_6T_4,\ C_7T_4,\ C_8T_4,\ T_1T_4 + T_4^2,\ T_2T_4 + T_4^2,\ T_3T_4,\\ &2C_1 - 2C_5 + 2C_6 + C_7 + C_8 + T_1 - 2T_2 - T_3 - T_4,\ C_3 + 2C_6 + C_8 - T_2 - T_3 - T_4,\\ &C_2 + 2C_5 - 2C_6 - C_8 - T_1 + T_2,\ 2C_4 + C_7 + C_8 - T_1 - T_3 - T_4). \end{aligned}$$

Moreover, we can compute a monomial basis of $\mathbb{Z}[C_1,\dots,C_8,T_1,\dots,T_4]/_I$ as a $\mathbb{Z}$-module (`SageMath` has a method `normal_basis` that is able to do so). It turns out that

$$\{T_4^2, T_4, T_3, T_2, T_1, C_8, C_7, C_6, C_5, 1\}$$

is a possible choice, therefore we conclude that

$$H^4(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}, \qquad H^2(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}^8, \qquad H^0(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}.$$

Remark. In this simple case one can check that the model $Y_{\mathcal{A}}$ is obtained as the blow-up of six points in $\mathbb{P}^1 \times \mathbb{P}^1$. The Betti numbers provided above agree with this construction.

Example 7.5. The next example is an arrangement defined by a matrix that has a non-primitive column:

$$\begin{pmatrix} 3 & 0 & 3 \\ 0 & 1 & -2 \end{pmatrix}.$$

This arrangement has five layers, and the first column gives rise to three of them. Its realization as arrangement in $(S^1)^2$ is shown in Figure 7.4. Figure 7.5 contains its poset of layers and the fan computed by `smooth_equal_sign_fan`. Recall that we chose to use the whole $\mathcal{C}_0(\mathcal{A})$ as the well-connected building set.

Using our algorithm, we find that the cohomology ring $H^*(Y_{\mathcal{A}};\mathbb{Z})$ is isomorphic to the quotient of $\mathbb{Z}[C_1,\dots,C_{10},T_1,\dots,T_{11}]$ by the ideal

$$\begin{aligned} I = (&T_{11}^3,\ C_6^2 - T_{11}^2,\ C_6C_7,\ C_7^2 - 3T_{11}^2,\ C_6C_8,\ C_7C_8,\ C_8^2 - 2T_{11}^2,\ C_6C_9,\ C_7C_9,\\ &C_8C_9 + T_{11}^2,\ C_9^2 - T_{11}^2,\ C_6C_{10} + T_{11}^2,\ C_7C_{10},\ C_8C_{10},\ C_9C_{10} + T_{11}^2,\ C_{10}^2 - 3T_{11}^2,\\ &C_6T_3,\ C_7T_3,\ C_8T_3,\ C_9T_3 + T_{11}^2,\ C_{10}T_3,\ T_3^2 - 3T_{11}^2,\ C_6T_4,\ C_7T_4,\ C_8T_4 + T_{11}^2, \end{aligned}$$

[8] This step is *not* part of the algorithm `wonderful_cohomology_ring`.

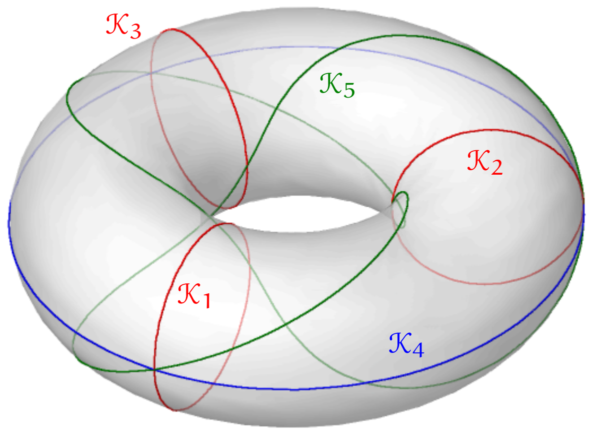


Figure 7.4: Arrangement with five layers, defined by the vectors $(3,0)$, $(0,1)$ and $(3,-2)$.

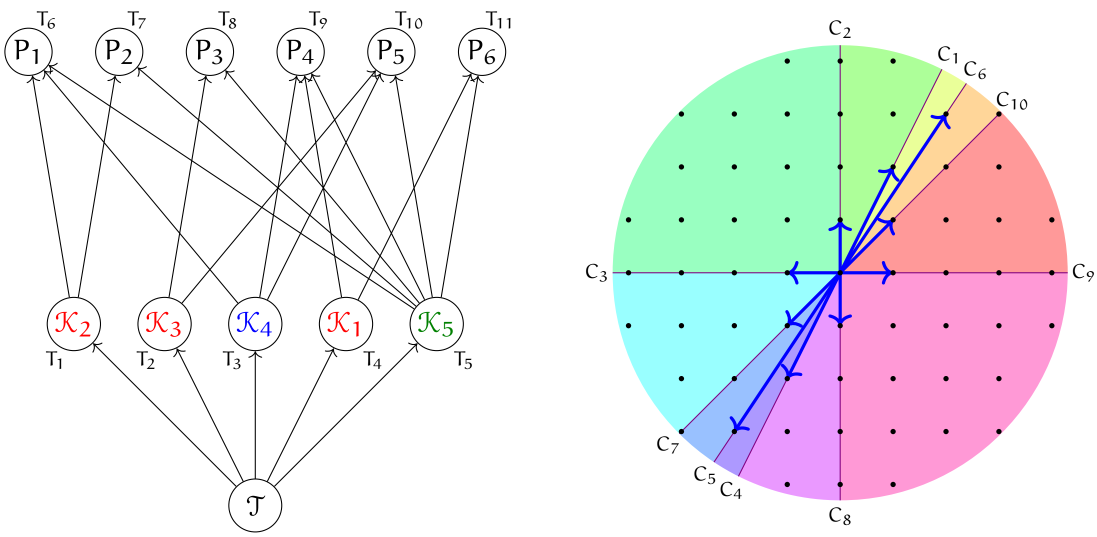


Figure 7.5: Poset of layers $\mathcal{C}(\mathcal{A})$ and fan Δ for the arrangement of Figure 7.4. The labels refer to the indeterminates associated with each element of $\mathcal{C}_0(\mathcal{A})$ and each ray of Δ.

$$C_9T_4,\ C_{10}T_4,\ T_3T_4,\ T_4^2-2T_{11}^2,\ C_6T_5+T_{11}^2,\ C_7T_5,\ C_8T_5,\ C_9T_5,\ C_{10}T_5,\ T_3T_5,$$
$$T_4T_5,\ T_5^2-6T_{11}^2,\ C_6T_6,\ C_7T_6,\ C_8T_6,\ C_9T_6,\ C_{10}T_6,\ T_3T_6+T_{11}^2,\ T_4T_6,\ T_5T_6+T_{11}^2,$$
$$T_6^2-T_{11}^2,\ C_6T_7,\ C_7T_7,\ C_8T_7,\ C_9T_7,\ C_{10}T_7,\ T_3T_7,\ T_4T_7,\ T_5T_7+T_{11}^2,\ T_6T_7,$$
$$T_7^2-T_{11}^2,\ C_6T_8,\ C_7T_8,\ C_8T_8,\ C_9T_8,\ C_{10}T_8,\ T_3T_8,\ T_4T_8,\ T_5T_8+T_{11}^2,\ T_6T_8,\ T_7T_8,$$
$$T_8^2-T_{11}^2,\ C_6T_9,\ C_7T_9,\ C_8T_9,\ C_9T_9,\ C_{10}T_9,\ T_3T_9+T_{11}^2,\ T_4T_9+T_{11}^2,\ T_5T_9+T_{11}^2,$$

$$T_6T_9,\ T_7T_9,\ T_8T_9,\ T_9^2 - T_{11}^2,\ C_6T_{10},\ C_7T_{10},\ C_8T_{10},\ C_9T_{10},\ C_{10}T_{10},\ T_3T_{10} + T_{11}^2,$$
$$T_4T_{10},\ T_5T_{10} + T_{11}^2,\ T_6T_{10},\ T_7T_{10},\ T_8T_{10},\ T_9T_{10},\ T_{10}^2 - T_{11}^2,\ C_6T_{11},\ C_7T_{11},$$
$$C_8T_{11},\ C_9T_{11},\ C_{10}T_{11},\ T_3T_{11},\ T_4T_{11} + T_{11}^2,\ T_5T_{11} + T_{11}^2,\ T_6T_{11},\ T_7T_{11},\ T_8T_{11},$$
$$T_9T_{11},\ T_{10}T_{11},\ C_2 - C_6 - 2C_9 - C_{10} - T_3 + 2T_4 - T_6 + T_9 - T_{10} + 2T_{11},$$
$$3C_3 + C_7 + 3C_9 + C_{10} + 2T_3 - 3T_4 - T_5 + T_6 - T_7 - T_8 - 2T_9 + T_{10} - 4T_{11},$$
$$C_4 + 2C_8 + 3C_9 + C_{10} - T_5 - T_6 - T_7 - T_8 - T_9 - T_{10} - T_{11},$$
$$3C_5 + C_7 - 3C_8 - 6C_9 - 2C_{10} - T_3 + 2T_5 + T_6 + 2T_7 + 2T_8 + T_9 + T_{10} + 2T_{11},$$
$$C_1 + 2C_6 + C_9 + C_{10} - T_4 - T_9 - T_{11},\ T_1 - T_4 + T_6 + T_7 - T_9 - T_{11},$$
$$T_2 - T_4 + T_8 - T_9 + T_{10} - T_{11}).$$

In particular the computation of a monomial basis gives us

$$H^4(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}, \qquad H^2(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}^{14}, \qquad H^0(Y_{\mathcal{A}};\mathbb{Z}) \simeq \mathbb{Z}.$$

Example 7.6. For the next example we bring back the characteristic varieties of hyperplane arrangements. Recall from Chapter 5 that the characteristic variety of the d$A3$ arrangement has one essential 2-dimensional component, given by

$$\{(t_1, \dots, t_5) \in (\mathbb{C}^*)^5 \mid t_1 - t_4 = 0,\ t_2 - t_5 = 0,\ t_3t_4t_5 - 1 = 0\}.$$

It is not hard to see that it is the intersection of the three layers

$$\begin{aligned} \mathcal{K}_1 &= \{(t_1, \dots, t_5) \in (\mathbb{C}^*)^5 \mid t_1t_4^{-1} = 1\}, \\ \mathcal{K}_2 &= \{(t_1, \dots, t_5) \in (\mathbb{C}^*)^5 \mid t_2t_5^{-1} = 1\}, \\ \mathcal{K}_3 &= \{(t_1, \dots, t_5) \in (\mathbb{C}^*)^5 \mid t_3t_4t_5 = 1\}. \end{aligned}$$

Therefore we consider the arrangement given by the matrix

$$M_{dA3} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 1 \\ 0 & -1 & 1 \end{pmatrix}.$$

Obviously we can draw neither the arrangement, nor the fan, but only the poset of layers (Figure 7.6), which has 8 elements. The fan computed by `dcg_algorithm` has 30 rays, therefore the cohomology ring is isomorphic to a quotient of

$$\mathbb{Z}[C_1, \dots, C_{30}, T_1, \dots, T_7]$$

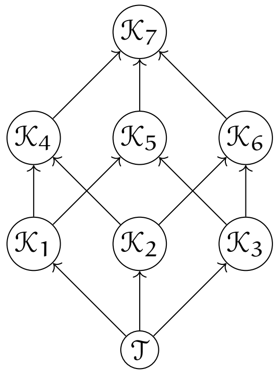


Figure 7.6: Poset of layers for the arrangement given by the matrix M_{dA3}.

by an ideal whose reduced DEGREVLEX Gröbner basis has 435 polynomials. We report here only the results of `normal_basis`, for the details refer to [10]:

$$\begin{aligned} H^{10}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}, \\ H^{8}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}^{29}, \\ H^{6}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}^{132}, \\ H^{4}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}^{132}, \\ H^{2}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}^{29}, \\ H^{0}(Y_{\mathcal{A}};\mathbb{Z}) &\simeq \mathbb{Z}. \end{aligned}$$

Example 7.7. In this last example we show the difference between a wonderful model computed using the whole $\mathcal{C}_0(\mathcal{A})$ as building set and the one computed using the minimal building set. We define the *toric braid arrangement* $T\mathcal{B}r_m$ as the arrangement in $(\mathbb{C}^*)^m$ with layers

$$\mathcal{K}_{ij} \coloneqq \{\mathbf{t} \in (\mathbb{C}^*)^m \mid t_i = t_j\}$$

for $1 \leqslant i < j \leqslant m$. The matrix associated with it is

$$\begin{pmatrix} 1 & 1 & \cdots & 1 & 0 & 0 & \cdots & 0 & \cdots & 0 \\ -1 & 0 & \cdots & 0 & 1 & 1 & \cdots & 1 & \cdots & 0 \\ 0 & -1 & \cdots & 0 & -1 & 0 & \cdots & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & -1 & \cdots & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots & \vdots & \vdots & \cdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 & \cdots & 1 \\ 0 & 0 & \cdots & -1 & 0 & 0 & \cdots & -1 & \cdots & -1 \end{pmatrix}.$$

We consider the arrangement $T\mathcal{B}r_4$ in $(\mathbb{C}^*)^4$ with six layers (higher dimensional ones require too much time to be computed). In Figure 7.7 we show the Hasse diagram

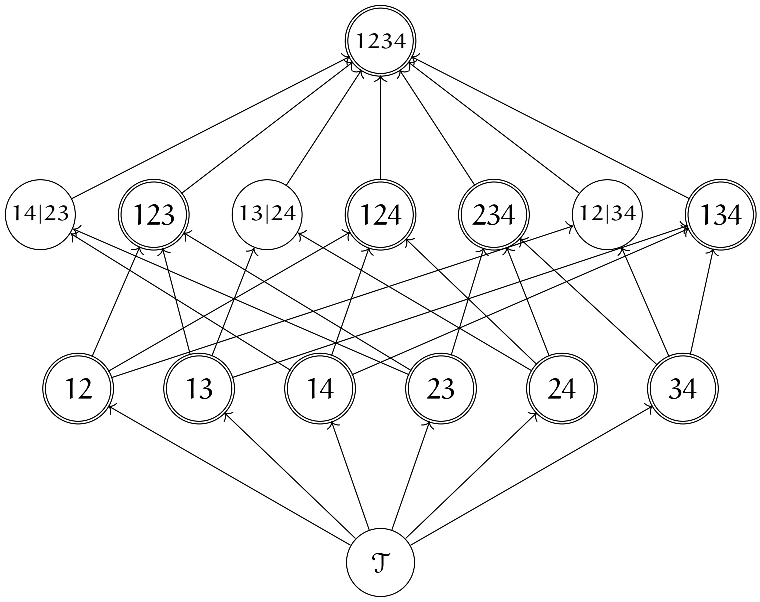


Figure 7.7: Poset of layers and minimal well-connected building set for the toric braid arrangement $T\mathcal{B}r_4$ in $(\mathbb{C}^*)^4$.

of its poset of layer; the labels of the vertices represent the groups of variables that are equal (so, for example, 12|34 represents the layer $\{\mathbf{t} \in (\mathbb{C}^*)^4 \mid t_1 = t_2,\ t_3 = t_4\}$). The double-circled vertices correspond to the minimal building set as computed by `minimal_well_connected_building_set`.

Let $Y_{\mathcal{A}}(\mathcal{C}_0)$ and $Y_{\mathcal{A}}(\mathcal{G}_{\min})$ be the wonderful models obtained by choosing as building sets the whole $\mathcal{C}_0(\mathcal{A})$ and the minimal building set respectively. We won't report the actual ideals here, because they are too big—they can be found in [10]; in the following table we report just the Betti numbers.

$i =$	0	2	4	6	8
$H^i(Y_{\mathcal{A}}(\mathcal{C}_0); \mathbb{Z})$	1	43	123	43	1
$H^i(Y_{\mathcal{A}}(\mathcal{G}_{\min}); \mathbb{Z})$	1	40	108	40	1

7.3 Towards a Generalization

The fact that we use Lenz's algorithm for the computation of the poset of layers forces us to deal only with arrangements that are compatible with Lenz's definition of "toric arrangement", i.e. divisorial ones of the form (7.2). In this section we try to modify some of the algorithms in order to allow for a more general type of toric arrangements.

The definition of layer (Definition 6.11) is rather difficult to work with, because of its abstractness. To overcome this problem, we fix coordinates $(t_1, \dots, t_n)$ on the ambient torus $\mathcal{T}$. This way we can identify $\mathcal{T}$ with $(\mathbb{C}^*)^n$ and the group of its character $X^*(\mathcal{T})$ with $\mathbb{Z}^n$. In particular, suppose that $\mathcal{K}(\Gamma, \varphi)$ is a layer and let $(\mathbf{a}_1, \dots, \mathbf{a}_m)$ be a basis of Γ, where

$$\mathbf{a}_i = (a_{i,1}, \dots, a_{i,n}) \in \mathbb{Z}^n;$$

then the layer is described by the system of equations

$$\begin{cases} t_1^{a_{1,1}} \cdots t_n^{a_{1,n}} = b_1, \\ \vdots \\ t_1^{a_{m,1}} \cdots t_n^{a_{m,n}} = b_m, \end{cases}$$

where $b_i = \varphi(\mathbf{a}_i) \in \mathbb{C}^*$. So we defined three classes to implement toric arrangements in `SageMath`: `ToricEquation`, `ToricLayer` and `ToricArrangement`.

Remark. Since the algorithms that we are going to outline depend on the ones in the previous sections, we cannot work with toric arrangements in full generality; in particular, *the only* b_i*'s that we allow are the roots of unity (of any order).*

```
class ToricEquation():
    def __init__(self,exponents,root=0):
        self._exponents=exponents
        self._root=root.numerator().mod(root.denominator())/root.denominato
        ↪ r()

    def exponents(self):
        return self._exponents

    def root(self):
        return self._root

    def root_order(self):
        if self._root==0:
            return 1
        else:
            return self._root.denominator()

    def root_exponent(self):
        if self._root==0:
            return 0
        else:
            return self._root.numerator()
```

```
23
24    def ambient_space_dimension(self):
25        return len(self._exponents)
```

An instance of a `ToricEquation` represents a single equation of the form

$$t_1^{a_1} \cdots t_n^{a_n} = b$$

with $a_1, \dots, a_n \in \mathbb{Z}$ and b is a root of unity; more precisely, `ToricEquation` needs two inputs:

1. a list of integers $[a_1, \dots, a_n]$, representing the exponents $a_1, \dots, a_n$;
2. a rational number r, with $0 \leqslant r < 1$, that represents the complex number $b = e^{2\pi i r}$.

The number r can be omitted, in which case it defaults to 0 (i.e. $b = 1$). Note that r could be any rational number, but since $e^{2\pi i r} = e^{2\pi i (r+k)}$ for any $k \in \mathbb{Z}$, the initialising procedure `__init__` always reduces it to the interval $0 \leqslant r < 1$ (see line 4).

Example 7.8. The call `ToricEquation([2,0,-1],1/2)` defines an object representing the equation

$$t_1^2 t_3^{-1} = -1.$$

The actual `ToricEquation` class defines also a procedure to check equality (two `ToricEquation`s are equal if and only if they have the same exponents and the same root). Moreover, it defines a procedure that allows us to compare two equations: we choose to set

$$\texttt{ToricEquation([}a_1, \dots, a_n\texttt{],r)} < \texttt{ToricEquation([}b_1, \dots, b_m\texttt{],s)}$$

if and only if $[a_1, \dots, a_n] < [b_1, \dots, b_m]$ in the lexicographic order,[9] or $[a_1, \dots, a_n] = [b_1, \dots, b_m]$ and $r < s$. This total ordering on the `ToricEquation`s is needed to define equality at `ToricLayer` level (see below).

The class `ToricEquation` defines methods to obtain the information stored in a `ToricEquation` instance. In the following list, we suppose that the methods are called on an object initialised as `ToricEquation(`$[a_1, \dots, a_n]$`,r)`.

`exponents()` Return the list $[a_1, \dots, a_n]$.

`root()` Return the number r.

[9] If $L_1 = [\ell_1, \dots, \ell_r]$ and $L_2 = [m_1, \dots, m_s]$ are two lists, then $L_1 < L_2$ in LEX order if and only if $\ell_1 < m_1$, or $\ell_1 = m_1$ and $\ell_2 < m_2$, or $\ell_1 = m_1$, $\ell_2 = m_2$ and $\ell_3 < m_3$, and so on. If the end of one list is reached without breaking the tie, the shorter list is the lesser one.

root_order() Return the order of the root of unity, i.e. if $r = p/q$ with $\mathrm{GCD}(p, q) = 1$, return q.

root_exponent() Return the exponent of the root of unity, i.e. if $r = p/q$ with $\mathrm{GCD}(p, q) = 1$, return p.

ambient_space_dimension() Return the length of $[a_1, \ldots, a_n]$, i.e. n; it is the dimension of the torus where the equation is defined.

The next class that we define is the one used to represent a single layer; we called it `ToricLayer`.

```
class ToricLayer():
    def __init__(self,equation_list):
        equation_list=sorted(list(Set(equation_list)))
        if Set([eq.ambient_space_dimension() for eq in equation_list]).cardinality()!=1:
            raise ValueError("equations have different lengths")
        M=matrix([eq.exponents() for eq in equation_list])
        eldiv=M.elementary_divisors()
        if not Set(eldiv).issubset(Set([0,1])):
            raise ValueError("equations do not define a connected layer")
        self._equations=equation_list
        self._ambient_space_dimension=equation_list[0].ambient_space_dimension()

    def equations(self):
        return self._equations

    def ambient_space_dimension(self):
        return self._ambient_space_dimension

    def n_equations(self):
        return len(self._equations)
```

A `ToricLayer` is just a list of `ToricEquations`. When an instance of `ToricLayer` is initialised, the procedure __init__ does some checks in order to make sure that the layer is well-defined:

1. first of all, the list of equations is sorted (this is possible because `ToricEquations` are comparable) and repeated equations are removed, if any;

2. then the procedure verifies that all the equations live in the same ambient space, i.e. if all `ambient_space_dimensions` are the same, and raises an error if this test fails;

3. after that it checks if the layer is connected by verifying that the lattice generated by all the `exponents` of the equations is a split direct summand of $\mathbb{Z}^n$; this is done by computing the elementary divisors of the matrix whose rows are the `exponents` (lines 6–7) and verifying that only zeros or ones appear among them (line 8).

A `ToricLayer` is defined only if the equations pass all these controls. Once an object of type `ToricLayer` is created, we can use the following methods on it.

`equations()` Return the list of `ToricEquations` that define the layer.

`ambient_space_dimension()` Return the dimension of the torus where the layer is defined; it is the `ambient_space_dimension` of any of the `ToricEquations` of the layer.

`n_equations()` Return the number of `ToricEquations` that define the layer.

Moreover, two `ToricLayers` are defined to be equal if and only if their (ordered) lists of equations are the same, and we say that

$$\texttt{ToricLayer([eq}_1,\dots,\texttt{eq}_h\texttt{])} < \texttt{ToricLayer([eq'}_1,\dots,\texttt{eq'}_k\texttt{])}$$

if and only if $\texttt{[eq}_1,\dots,\texttt{eq}_h\texttt{]} < \texttt{[eq'}_1,\dots,\texttt{eq'}_k\texttt{]}$ in the lexicographic order.

The last class that we define is `ToricArrangement`, which allows us to represent toric arrangements.

```
class ToricArrangement():
   def __init__(self,layer_list):
      if Set([layer.ambient_space_dimension() for layer in layer_list]).cardinality()!=1:
         raise ValueError("layers live in different spaces")
      self._layers=sorted(list(Set(layer_list)))
      self._ambient_space_dimension=layer_list[0].ambient_space_dimension()

   def __add__(self,other):
      return ToricArrangement(self._layers+other._layers)

   def layers(self):
```

```
12          return self._layers
13
14      def ambient_space_dimension(self):
15          return self._ambient_space_dimension
16
17      def n_layers(self):
18          return len(self._layers)
```

A `ToricArrangement` object is defined by a list of `ToricLayers`. Before creating a `ToricArrangement`, the initialising procedure checks if all the layers belong to the same ambient torus, and raises an error if this control fails. If the layers are compatible, the procedure sorts the list and removes repetitions before defining the arrangement (line 5).

Two `ToricArrangement` objects are considered equal if and only if they have the same (ordered) list of `ToricLayers` (this is why we define a comparison procedure within the class `ToricLayer`). We defined also a sum of arrangements (method `__add__` above): if `A1` and `A2` are two instances of `ToricArrangement`, then `A1+A2` is the arrangement defined by the union of the layers of `A1` and `A2`.

As before, we define some usual methods to retrieve information from an object of type `ToricArrangement`.

`layers()` Return the list of `ToricLayers` that define the arrangement.

`ambient_space_dimension()` Return the dimension of the torus where the arrangement is defined; it is the `ambient_space_dimension` of any of the `ToricLayers` of the arrangement.

`n_layers()` Return the number of `ToricLayers` that define the arrangement.

In order to use the algorithms given in the previous sections with the objects of type `ToricArrangement` that we just defined, we need to recover a matrix that describes the arrangement. The idea is the following: if

$$t_1^{a_1} \cdots t_n^{a_n} = (\zeta_q)^p \tag{7.9}$$

is an equation of a layer of an arrangement $\mathcal{A}$, where $\zeta_q = e^{2\pi i/q}$ and $0 \leqslant p < q$, we put the list $(q\,a_1, \dots, q\,a_n)$ as a column of the matrix.[10] By juxtaposing the lists obtained from all the equations of all layers in $\mathcal{A}$, we get a matrix M. Now, if we consider the arrangement $\mathcal{A}_M$ defined by M as in Equation (7.2), we have that each

[10] Actually, there is the possibility that two equations have the same exponents with roots of different orders. We make sure that the corresponding columns are added to the matrix only once, taking care of the orders.

layer of $\mathcal{A}$ can be obtained as intersection of some of the 1-codimensional layers of $\mathcal{A}_M$. We keep track of the original layers of $\mathcal{A}$ by building a list while defining the matrix M: the i-th element of this list represents the i-th layer of $\mathcal{A}$, and it is a list itself, whose j-th element is a pair (k, p) that represents an equation, where k is the index of the column in M corresponding to the exponents of the equation, and p is the root exponent as in Equation (7.9).

We encoded the procedure just outlined in a function called `matrix_from_toric`, which we don't report here.

Example 7.9. Let $\mathcal{A}$ be the arrangement of $(\mathbb{C}^*)^3$ with the three layers

$$L_1\colon \left\{ t_1^2 t_3 = 1, \right. \qquad L_2\colon \begin{cases} t_1 t_2^{-1} = 1, \\ t_1 t_3 = (\zeta_3)^2, \end{cases} \qquad L_3\colon \begin{cases} t_1 t_2^{-1} = 1, \\ t_1^{-1} t_2^{-2} t_3 = 1. \end{cases}$$

Then the corresponding matrix is

$$M = \begin{pmatrix} 2 & 1 & 3 & -1 \\ 0 & -1 & 0 & -2 \\ 1 & 0 & 3 & 1 \end{pmatrix}$$

and the layers are described by the list[11]

$$\Big[\big[(0,0)\big], \big[(1,0),(2,2)\big], \big[(1,0),(3,0)\big] \Big].$$

It is time to see how we modify the algorithms so that they work with this new setting. As far as the good toric variety $X_{\mathcal{A}}$ is concerned, no adjustment is needed: we apply either `dcg_algorithm` or `smooth_equal_sign_fan` (depending on the dimension of the arrangement) to the set Ξ given by the columns of the matrix obtained from `matrix_from_toric`. The poset of layers is a little trickier, so we report the actual code here and comment it.

```
def poset_of_layers(arr):
   M,L=matrix_from_toric(arr)
   P=poset_of_layers_lenz(M)
   red=[]
   for p in P[1:]:
      minimal=[lay for lay in P.level_sets()[1] if P.is_lequal(lay,p)]
      description=sorted([(lay[0][0],lay[1].lift()[0]) for lay in
       ↪  minimal])
      if description in L:
         red+=[p]
```

[11] Recall that `SageMath` indexes the list (including the rows and columns of matrices) starting from 0.

```
10    for level in P.level_sets()[2:]:
11       for p in level:
12          if p not in red:
13             color=[c for c in red if P.is_lequal(c,p)]
14             if len(color)==0:
15                continue
16             Int=intersection_of_poset_subset(P,color)
17             if p in Int:
18                red+=[p]
19    return P.subposet([P[0]]+red)
```

At first the algorithm computes the matrix `M` and the list `L` as described previously, then it computes the poset of layers of the arrangement $\mathcal{A}_M$ using Lenz's algorithm (line 3).

The poset of layers of `arr` is a subposet of $P = \mathcal{C}(\mathcal{A}_M)$: we have to identify its elements. The algorithm will "colour" them in red, so it creates a list `red` that will contain them (line 4).

First of all, using the information of the list `L`, the algorithm identifies the elements of `P` that correspond to the layers of `arr` (lines 5–9), which obviously have to belong to $\mathcal{C}(\texttt{arr})$. For each element $p \in P$ (excluding the whole torus), the algorithm computes the layers of $\mathcal{A}_M$ such that `p` belongs to their intersection (line 6); in line 7 the algorithm computes a description of `p` as a list of pairs which is comparable to the ones in `L`—in other words, if $\texttt{description} = [(i_1, p_1), \dots, (i_k, p_k)]$, then `p` is obtained as the intersection of the $(p_j + 1)$-th component among the ones defined by the $(i_j + 1)$-th column, for $j = 1, \dots, k$. The algorithm then colours the `p`'s whose `description` is in `L` (lines 8–9).

The next step is to identify the components corresponding to the intersections of layers of `arr` (lines 10–18); the algorithm does this inductively, one level at a time. Let `p` be an element of `P` and suppose that for each element `q` that belongs to a lesser level than `p` we have already decided whether `q` is red or not. If `p` is already red, the algorithm continues; otherwise, it computes the set of the red elements which are less than or equal to `p` (that is to say, the red layers that contain it). If `p` is contained in no red layer, the algorithm discards it (lines 14–15); otherwise, it computes the intersection of those layers and verifies that `p` is indeed a connected component of this intersection, eventually adding it to the `red` list.

Example 7.10. Consider the arrangement $\mathcal{A}$ in $(\mathbb{C}^*)^3$ defined by the layers

$$L_1 \colon \left\{ t_2^2 t_3^{-3} = 1, \right. \qquad L_2 \colon \left\{ t_1 = 1, \right. \qquad L_3 \colon \begin{cases} t_1 t_3 = -1, \\ t_1^2 t_2 = 1. \end{cases}$$

The matrix obtained from `matrix_from_toric` is

$$M = \begin{pmatrix} 0 & 1 & 2 & 2 \\ 2 & 0 & 0 & 1 \\ -3 & 0 & 2 & 0 \end{pmatrix}$$

which defines an arrangement $\mathcal{A}_M$ with five layers

$$\begin{aligned} E_1 &: t_2^2 t_3^{-3} = 1, \\ E_2 &: t_1 = 1, \\ E_3 &: t_1 t_3 = 1, \\ E_4 &: t_1 t_3 = -1, \\ E_5 &: t_1^2 t_2 = 1. \end{aligned}$$

In Figure 7.8 on the left we have the poset of layers of $\mathcal{A}_M$. The layers of $\mathcal{A}$ can be identified easily in this poset:

$$\begin{aligned} L_1 &\leftrightarrow E_1, \\ L_2 &\leftrightarrow E_2, \\ L_3 &\leftrightarrow H_9 = E_4 \cap E_5. \end{aligned}$$

The result of the algorithm is the subposet given by the highlighted elements, which is drawn more clearly on the right of Figure 7.8.

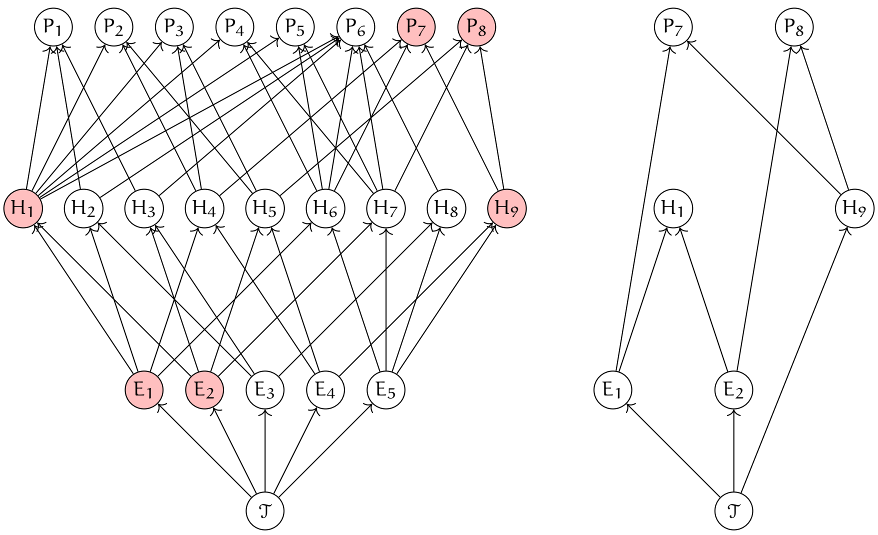


Figure 7.8: Posets for the arrangement of Example 7.10. On the left: poset of layers of $\mathcal{A}_M$, with layers of $\mathcal{C}(\mathcal{A})$ highlighted. On the right: poset of layers of $\mathcal{A}$.

The last remarkable difference between the algorithms of Section 7.1 and this one is in the `minimal_well_connected_building_set` algorithm (see page 113). In line 3 of that algorithm we initialise `G` as `P.level_sets()[1]`, because that level set contains exactly the layers of the arrangement. This is no longer true in general: for example it can happen that one layer of the toric arrangement is contained in another one (for example, an arrangement may have both $\{t_1 = 1, t_2 = 1\}$ and $\{t_1 = 1\}$ as layers), so that the second layer belongs to a higher level set.

The solution actually is quite easy: we have to identify the layers of the original arrangement. Lines 4–9 of the algorithm `poset_of_layers` do exactly that, so we change line 3 of `minimal_well_connected_building_set` with those lines.

Once that we have the fan associated with the variety $X_{\mathcal{A}}$, the poset of layers $\mathcal{C}(\mathcal{A})$ and a (minimal) well-connected building set $\mathcal{G}$, we can use the algorithm `wonderful_cohomology_ring` almost as it is, only with small technical adjustments (for example in the `bases_dictionary` procedure), because those are the only information needed by that algorithm.

Example 7.11. Let's get back to the arrangement of Example 7.10. We want to compute the presentation of $H^*(Y_{\mathcal{A}}; \mathbb{Z})$ where $Y_{\mathcal{A}}$ is built using the minimal well-connected building set. With reference to Figure 7.8, it is not hard to see that this building set is $\mathcal{G} = \{E_1, E_2, H_9\}$: the three layers of $\mathcal{A}$ have to be included in $\mathcal{G}$ and their intersections are all connected, so they are included in $\mathcal{G}$ if and only if they are *not* transversal. In this case

$$\begin{aligned}
\operatorname{codim}(H_1) &= 2 = 1 + 1 = \operatorname{codim}(E_1) + \operatorname{codim}(E_2),\\
\operatorname{codim}(P_7) &= 3 = 1 + 2 = \operatorname{codim}(E_1) + \operatorname{codim}(H_9),\\
\operatorname{codim}(P_8) &= 3 = 1 + 2 = \operatorname{codim}(E_2) + \operatorname{codim}(H_9),
\end{aligned}$$

so no other element of $\mathcal{C}_0(\mathcal{A})$ belongs to $\mathcal{G}$. After a little computation, we get that

- the fan Δ computed using `dcg_algorithm` has 34 rays, therefore $H^*(Y_{\mathcal{A}}; \mathbb{Z})$ is a quotient of $\mathbb{Z}[C_1, \ldots, C_{34}, T_1, T_2, T_3]$;
- a reduced Gröbner basis of the relations ideal has 522 polynomials (we won't write them here, check [10] if you want to see them);
- the Betti numbers of $Y_{\mathcal{A}}$ are

$$H^6(Y_{\mathcal{A}}; \mathbb{Z}) \simeq \mathbb{Z}, \quad H^4(Y_{\mathcal{A}}; \mathbb{Z}) \simeq \mathbb{Z}^{32}, \quad H^2(Y_{\mathcal{A}}; \mathbb{Z}) \simeq \mathbb{Z}^{32}, \quad H^0(Y_{\mathcal{A}}; \mathbb{Z}) \simeq \mathbb{Z}.$$

In conclusion, with the algorithms outlined in this chapter we are able to compute projective wonderful models (using arbitrary building sets) and their integer cohomology rings for toric arrangements with layers of any codimension, when the equations involved have the form

$$t_1^{a_1} \cdots t_n^{a_n} = \zeta$$

where ζ is any root of unity. There is a limitation, though: even not so big arrangements may give rise to ideals with a huge number of generators (for example, we tried to compute the cohomology of $Y_{\mathcal{A}}$ for $\mathcal{A} = \mathrm{T}\mathcal{B}r_5$,[12] but the relations involved were more than 33 millions and our computer did not manage to finish the computation). In the future we hope to improve our algorithms to make them more efficient from this point of view.

[12] See Example 7.7 for the definition of the toric braid arrangement $\mathrm{T}\mathcal{B}r_m$.

Bibliography

[1] Donu Arapura. “Geometry of cohomology support loci for local systems I”. In: *Jour. Algebraic Geom.* 6.3 (1997).

[2] Enrique Artal et al. “Invariants of Combinatorial Line Arrangements and Rybnikov’s Example”. In: *Singularity Theory and Its Applications*. Vol. 43. Advanced Studies in Pure Mathematics. Tokyo: Math. Soc. Japan, 2006, pp. 1–43.

[3] Pauline Bailet and Simona Settepanella. “Homology graph of real arrangements and monodromy of Milnor fiber”. In: *Adv. Appl. Math.* 90 (2017), pp. 46–85.

[4] Anders Björner et al. *Oriented Matroids*. 2nd ed. Encyclopedia of Mathematics and its Applications. Cambridge Univ. Press, 1999.

[5] Nicolas Bourbaki. *Lie Groups and Lie Algebras. Chapter 4–6*. Elements of Mathematics. Berlin Heidelberg: Springer-Verlag, 2002.

[6] Filippo Callegaro and Emanuele Delucchi. “The integer cohomology algebra of toric arrangements”. In: *Advances in Mathematics* 313 (2017), pp. 746–802.

[7] Daniel C. Cohen. “Triples of Arrangements and Local Systems”. In: *Proc. Amer. Math. Soc.* 130.10 (2002), pp. 3025–3031.

[8] Daniel C. Cohen and Alexander I. Suciu. “Characteristic Varieties of Arrangements”. In: *Math. Proc. Cambridge Phil. Soc.* 127.1 (1999), pp. 33–53.

[9] Daniel C. Cohen and Alexander I. Suciu. “On Milnor Fibrations of Arrangements”. In: *Jour. London Math. Soc.* 51.1 (1995), pp. 105–119.

[10] *Cohomology Rings of Projective Wonderful Models for Toric Arrangements*. URL: `http://people.dm.unipi.it/papini/ToricComputations.html`.

[11] Corrado De Concini and Giovanni Gaiffi. *Cohomology Rings of Compactifications of Toric Arrangements*. 2018. arXiv: `1801.04383v1 [math.AT]`.

[12] Corrado De Concini and Giovanni Gaiffi. “Projective wonderful models for toric arrangements”. In: *Advances in Mathematics* 327 (2018), pp. 390–409.

[13] Corrado De Concini and Claudio Procesi. “Hyperplane Arrangements and Holonomy Equations”. In: *Selecta Mathematica, New Series* 1.3 (1995), pp. 495–535.

[14] Corrado De Concini and Claudio Procesi. "On the Geometry of Toric Arrangements". In: *Transformation Groups* 10.3 (2005), pp. 387–422.

[15] Corrado De Concini and Claudio Procesi. "Wonderful Models of Subspace Arrangements". In: *Selecta Mathematica, New Series* 1.3 (1995), pp. 459–494.

[16] David A. Cox, John B. Little, and Henry K. Schenck. *Toric Varieties*. Providence, RI: Amer. Math. Soc., 2011.

[17] Michael Cuntz. "Simplicial Arrangements with up to 27 Lines". In: *Discrete Comput. Geom.* 48 (2012), pp. 682–701.

[18] Giacomo d'Antonio and Emanuele Delucchi. "A Salvetti Complex for Toric Arrangements and its Fundamental Group". In: *International Mathematics Research Notices* 15 (2012), pp. 3535–3566.

[19] Alexandru Dimca. *Hyperplane Arrangements. An Introduction*. Springer, 2017.

[20] Alexandru Dimca. "Pencils of Plane Curves and Characteristic Varieties". In: *Arrangements, Local Systems and Singularities*. Vol. 283. Progress in Math. Basel: Birkhäuser Verlag, 2010, pp. 59–82.

[21] Alexandru Dimca and Stefan Papadima. "Hypersurface complements, Milnor fibers and higher homotopy groups of arrangments". In: *Annals of Math.* 158.2 (2003), pp. 473–507.

[22] Meital Eliyahu, David Garber, and Mina Teicher. "A conjugation-free geometric presentation of fundamental groups of arrangements". In: *manuscripta mathematica* 133.1 (2010), pp. 247–271.

[23] Michael Falk. "Arrangements and Cohomology". In: *Annals of Combinatorics* 1 (1997), pp. 135–157.

[24] Michael Falk and Sergey Yuzvinsky. "Multinets, resonance varieties, and pencils of plane curves". In: *Compos. Math.* 143.4 (2007), pp. 1069–1088.

[25] William Fulton. *Introduction to Toric Varieties*. Vol. 131. Annals of Math. Studies. Princeton Univ. Press, 1993.

[26] Giovanni Gaiffi and Mario Salvetti. "The Morse Complex of a Line Arrangement". In: *Journal of Algebra* 321 (2009), pp. 316–337.

[27] Jacob E. Goodman and Richard Pollack. "Proof of Grünbaum's conjecture on the stretchability of certain arrangements of pseudolines". In: *Journal of Combinatorial Theory, Series A* 29.3 (1980), pp. 385–390.

[28] Branko Grünbaum. "A Catalogue of Simplicial Arrangements in the Real Projective Plane". In: *Ars Math. Contemp.* 2 (2009), pp. 1–25.

[29] Branko Grünbaum. *Arrangements and Spreads*. Vol. 10. CBMS Regional Conf. Series in Math. Providence, RI: Amer. Math. Soc., 1972.

[30] Allen Hatcher. *Algebraic Topology*. Cambridge Univ. Press, 2002.

[31] Matthias Lenz. *Computing the Poset of Layers of a Toric Arrangement*. 2017. arXiv: 1708.06646v2 [math.CO].

[32] Li Li. "Wonderful compactification of an arrangement of subvarieties". In: *Michigan Math. Jour.* 58.2 (2009), pp. 535–563.

[33] Anatoly Libgober. "Characteristic Varieties of Algebraic Curves". In: *Applications of Algebraic Geometry to Coding Theory, Physics and Computation*. Vol. 36. NATO Science Series II. Kluwer Academic Publishers, 2001, pp. 215–254.

[34] Anatoly Libgober and Sergey Yuzvinsky. "Cohomology of the Orlik-Solomon Algebras and Local Systems". In: *Compos. Math.* 121.3 (2000), pp. 337–361.

[35] John Milnor. *Singular Points of Complex Hypersurfaces*. Vol. 61. Annals of Math. Studies. Princeton Univ. Press, 1968.

[36] Luca Moci. "Wonderful Models for Toric Arrangements". In: *International Mathematics Research Notices* 1 (2012), pp. 213–238.

[37] Peter Orlik and Hiroaki Terao. *Arrangements of Hyperplanes*. Berlin Heidelberg: Springer-Verlag, 1992.

[38] Roberto Pagaria. *Combinatorics of Toric Arrangements*. 2017. arXiv: 1710.00409v1 [math.AT].

[39] Grigory Rybnikov. "On the Fundamental Group of the Complement of a Complex Hyperplane Arrangement". In: *Funct. Anal. Appl.* 45.2 (2011), pp. 137–148.

[40] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 8.0)*. http://www.sagemath.org. 2018.

[41] Mario Salvetti. "Topology of the Complement of Real Hyperplanes in $\mathbb{C}^N$". In: *Invent. Math.* 88 (1987), pp. 603–618.

[42] Mario Salvetti and Matteo Serventi. "Twisted Cohomology of Arrangements of Lines and Milnor Fibers". In: *Annali della Scuola Normale Superiore di Pisa, Classe di Scienze* XVII.4 (2017), pp. 1461–1489.

[43] Mario Salvetti and Simona Settepanella. "Combinatorial Morse Theory and Minimality of Hyperplane Arrangements". In: *Geometry & Topology* 11 (2007), pp. 1733–1766.

[44] Matteo Serventi. "Combinatorial and Geometric Invariants of Configuration Spaces". PhD thesis. Università di Pisa, 2015.

[45] Alexander I. Suciu. "Fundamental groups of line arrangements: Enumerative aspects". In: *Advances in Algebraic Geometry Motivated by Physics*. Vol. 276. Contemp. Math. Providence, RI: Amer. Math. Soc., 2001, pp. 43–79.

[46] Alexander I. Suciu. "Translated tori in the characteristic varieties of complex hyperplane arrangements". In: *Topology and its Appl.* 118 (2002), pp. 209–223.

[47] Masahiko Yoshinaga. "Milnor Fibers of Real Line Arrangements". In: *Jour. of Singularities* 7 (2013), pp. 220–237.

[48] Masahiko Yoshinaga. "Resonant Bands and Local System Cohomology Groups for Real Line Arrangements". In: *Vietnam Jour. Math.* 42.3 (2014), pp. 377–392.

Acknowledgements

Arrivati alla fine di questo percorso di dottorato, è doveroso ricordare tutte le persone che mi hanno accompagnato durante questi anni. Anzi, *ricordare* è un termine troppo riduttivo; sarebbe il caso di dire *ringraziare*, o meglio ancora *riconoscere*[13] con gratitudine il grande contributo umano e materiale di queste persone, senza il quale gli anni di dottorato sarebbero trascorsi assai diversamente.

In primo luogo un ringraziamento va ai relatori di questa tesi, prof. Mario Salvetti e prof. Giovanni Gaiffi, per il continuo sostegno e le discussioni costruttive in questi anni. Spero che questa non sia solo la fine di un ciclo, ma possa evolvere in una fruttuosa collaborazione. Grazie anche ai *referee* per le preziose osservazioni.

Il Dipartimento di Matematica dell'Università di Pisa è stata una seconda casa per dieci lunghi anni, e lo è diventato ancora di più quando, da dottorando, ho avuto a disposizione una scrivania in uno degli studi. Ma una casa non è fatta solo dall'edificio in sé; le persone che la abitano sono altrettanto importanti. Per questo ringrazio, senza fare la lista dei nomi, tutti gli amici e i colleghi del dipartimento: sempre pronti a scambiare due parole, siano esse chiacchiere davanti alle macchinette di caffè oppure tentativi di smontare teoremi particolarmente ostici.

D'altra parte, la vita universitaria non è fatta solo di "studio matto e disperatissimo" a casa o in dipartimento. Il cervello ha bisogno di staccare dai libri ogni tanto e variegare le sue attività: in caso contrario, a mio parere, perde di efficienza. Grazie alla mia ragazza sono entrato nel giro dell'Orchestra dell'Università di Pisa, non come musicista (d'altronde non metto mano seriamente al pianoforte da circa quindici anni) ma come collaboratore. Dunque ringrazio tutti gli amici dell'Orchestra e il suo direttore, Manfred, per i bei momenti durante le prove (sia con che senza strumenti).

I veri Atlanti che hanno sostenuto il mio mondo in questi anni sono stati i miei genitori, Floriana e Maurizio. Senza il loro incoraggiamento costante e la loro presenza, in particolare nei momenti di sconforto, non sarei mai arrivato alla fine di questo viaggio. Non credo che le poche parole scritte qui siano sufficienti per esprimere la mia gratitudine, ma spero di trovare anche altri mezzi per manifestarla.

È già stata citata *en passant* in questi ringraziamenti, ma merita certamente un

[13] Dopotutto, la traduzione migliore per *acknowledgements* è "riconoscimenti".

paragrafo a parte la mia ragazza Luigia. Sempre al mio fianco da più di cinque anni, ma sembra nemmeno un mese, ha illuminato le mie giornate, anche quelle più plumbee. Non dimenticherò mai le nostre uscite al mercatino di libri usati, il tempo passato a sistemare i faldoni con le parti per l'Orchestra, e anche le ore di studio silenzioso insieme. Grazie, di tutto.