

Invited talks

A New Symbolic Method for Linear Boundary Value Problems Using Groebner Bases

MARKUS ROSENKRANZ¹

(JOINT WORK WITH BRUNO BUCHBERGER²,
GEORG REGENSBURGER¹, AND LOREDANA TEC²)

¹ Johann Radon Institute for Computational and Applied Mathematics,
Austrian Academy of Sciences, Altenberger Str. 69, 4040 Linz, Austria, Europe
Markus.Rosenkranz@oeaw.ac.at, Georg.Regensburger@oeaw.ac.at

² Research Institute for Symbolic Computation,
Johannes Kepler University, 4032 Castle of Hagenberg, Austria, Europe
Bruno.Buchberger@risc.uni-linz.ac.at, ltec@risc.uni-linz.ac.at

Abstract

Boundary value problems are of utmost importance for science and engineering. In fact, most differential equations come along with boundary conditions of some sort. It is therefore surprising that such problems—even in the linear case—have gained little attention in Symbolic Computation. Consequently, their coverage in computer algebra systems is rather unsystematic and unpredictable.

The proper consideration of boundary conditions leads to a substantial revision of the algebraic structures currently used in established symbolic methods like differential algebra or differential Galois theory. One important ingredient in an algebraic approach to boundary value problems is the interaction of differential, integral and boundary operators. We present one such approach, based on Buchberger's powerful concept of Groebner bases.

For the implementation of the method we use the functor concept introduced by Buchberger for the Theorema system. This allows for easy adjustment of the code to various coefficient domains and different representations of the underlying objects.

Holonomic functions revisited

TOSHINORI OAKU

Tokyo Woman's Christian University
2-6-1, Zempukuji, Suginami-ku, Tokyo, 167-8585 Japan
oaku@lab.twcu.ac.jp

Abstract

A holonomic function is a differentiable or generalized function which satisfies a holonomic system of linear partial or ordinary differential equations with polynomial coefficients. We present algorithms for computing systems of differential equations which the sum, the product, the restriction, and the integration of holonomic functions satisfy. These algorithms are based on Gröbner bases of differential operators and are rigorous in the sense that the output differential equations are also holonomic.

1 Differential operators and holonomic systems

Let us denote by D_n the ring of differential operators on the variables $x = (x_1, \dots, x_n)$ with polynomial coefficients. An element P of D_n is written in a finite sum

$$P = \sum_{\alpha \in \mathbb{N}^n} a_{\alpha, \beta} x^\alpha \partial^\beta, \quad (1)$$

where $\alpha = (\alpha_1, \dots, \alpha_n)$, $\beta = (\beta_1, \dots, \beta_n) \in \mathbb{N}^n$ are vectors of nonnegative integers with $\mathbb{N} = \{0, 1, 2, \dots\}$, $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$, $\partial^\beta = \partial_1^{\beta_1} \cdots \partial_n^{\beta_n}$ with the derivations $\partial_i = \partial/\partial x_i$ ($i = 1, \dots, n$), and $a_{\alpha, \beta}$ are complex numbers.

Given $P_1, \dots, P_r \in D_n$, we associate the left ideal $I = D_n P_1 + \cdots + D_n P_r$ generated by P_1, \dots, P_r with a system of linear differential equations

$$P_1 u = \cdots = P_r u = 0 \quad (2)$$

for an unknown function u . This enables us to work with a left ideal of D_n instead of each system of linear differential equations. Here we suppose that the unknown function u belongs to a 'function space' \mathcal{F} which is a left D_n -module. Examples of such \mathcal{F} are the set $C^\infty(U)$ of C^∞ functions on an open subset U of \mathbb{R}^n , the set $\tilde{\mathcal{O}}(U)$ of possibly multi-valued analytic functions on an open subset U of \mathbb{C}^n , the set $\mathcal{D}'(U)$ of the Schwartz distributions on an open subset U of \mathbb{R}^n , and the set $S'(\mathbb{R}^n)$ of tempered distributions.

A weight vector for D_n is an integer vector

$$w = (w_1, \dots, w_n; w_{n+1}, \dots, w_{2n}) \in \mathbb{Z}^{2n}$$

with the conditions $w_i + w_{n+i} \geq 0$ for $i = 1, \dots, n$, which are necessary in view of the commutation relation $\partial_i x_i = x_i \partial_i + 1$ in D_n . For a nonzero differential operator P of the form (1), we define its w -order to be

$$\text{ord}_w(P) := \max\{\langle w, (\alpha, \beta) \rangle = w_1 \alpha_1 + \cdots + w_n \alpha_n + w_{n+1} \beta_1 + \cdots + w_{2n} \beta_n \mid a_{\alpha, \beta} \neq 0\},$$

and its w -initial part to be

$$\text{in}_w(P) := \sum_{\langle w, (\alpha, \beta) \rangle = \text{ord}_w(P)} a_{\alpha, \beta} x^\alpha \partial^\beta.$$

In particular, when $w = (\mathbf{0}, \mathbf{1}) = (0, \dots, 0; 1, \dots, 1)$, then the polynomial

$$\sigma(P)(x, \xi) := \sum_{\langle w, (\alpha, \beta) \rangle = \text{ord}_w(P)} a_{\alpha, \beta} x^\alpha \xi^\beta \in \mathbb{C}[x, \xi]$$

is called the *principal symbol* of P .

Definition 1 A left ideal I of D_n is said to be *holonomic* if the ideal $\sigma(I)$ of $\mathbb{C}[x, \xi]$ which are generated by the set $\{\sigma(P)(x, \xi) \mid P \in I, P \neq 0\}$ is of dimension n , that is, the characteristic variety of I , which is defined to be

$$\text{Char}(I) := \{(x, \xi) \in \mathbb{C}^{2n} \mid p(x, \xi) = 0 \text{ for any } p \in \sigma(I)\},$$

is of dimension n . (In general, the dimension of $\text{Char}(I)$ is greater than or equal to n if $I \neq D_n$.) We call (2) a *holonomic system* if the left ideal $I = D_n P_1 + \dots + D_n P_n$ is holonomic. We also call $\text{Char}(I)$ the characteristic variety of the holonomic system (2).

The characteristic variety of (2) can be computed by Gröbner bases ([4]). The dimension of the characteristic variety can be computed by using the Hilbert function.

2 Holonomic functions

Definition 2 Let u be a C^∞ function or a distribution (in the sense of L. Schwartz, or a generalized function in the sense of Gelfand-Shilov) defined on an open subset U of \mathbb{R}^n . Then we call u a *holonomic function* or a *holonomic distribution* on U if u satisfies a holonomic system. In other words, u is holonomic if and only if its *annihilator*

$$\text{Ann}_{D_n} u := \{P \in D_n \mid Pu = 0 \text{ on } U\}$$

is a holonomic ideal.

For example, given an arbitrary polynomial f , the C^∞ function e^f is holonomic on \mathbb{R}^n . In fact we can easily verify that

$$\text{Ann}_{D_n} e^f = D_n \left(\partial_1 - \frac{\partial f}{\partial x_1} \right) + \dots + D_n \left(\partial_n - \frac{\partial f}{\partial x_n} \right),$$

which shows that the characteristic variety is $\{(x, \xi) \in \mathbb{C}^{2n} \mid \xi_1 = \dots = \xi_n = 0\}$.

If f_1, \dots, f_m are nonzero polynomials with real coefficients, then the distribution $u = (f_1)_+^{\lambda_1} \cdots (f_m)_+^{\lambda_m}$ defined by

$$\langle u, \varphi \rangle = \int_{f_1 \geq 0, \dots, f_m \geq 0} f_1(x)^{\lambda_1} \cdots f_m(x)^{\lambda_m} \varphi(x) dx \quad (\varphi \in C_0^\infty(\mathbb{R}^n))$$

is holonomic on \mathbb{R}^n unless $(\lambda_1, \dots, \lambda_m) \in \mathbb{C}^m$ is not contained in an exceptional set where u cannot be defined. There is an algorithm to compute a holonomic ideal of which u is a solution if the coefficients of f_i are contained in a computable field. In particular, the product of Heaviside's functions

$$Y(f_1) \cdots Y(f_m) = (f_1)_+^0 \cdots (f_m)_+^0$$

is a holonomic distribution on \mathbb{R}^n . Elementary functions are not necessarily holonomic. For example, the smooth (C^∞) function x^y in the two variables x and y defined on $\{(x, y) \in \mathbb{R}^2 \mid x > 0, y \in \mathbb{R}\}$ is not holonomic in the sense above.

3 Operations on holonomic functions

For the sake of simplicity, let us assume that u and v are holonomic functions or distributions defined on the whole \mathbb{R}^n . Then the following functions or distributions are holonomic under the condition that they are well-defined. Moreover, if a holonomic ideal for u (and a one for v if relevant) is explicitly given, then there exist algorithms to compute a holonomic ideal which each of the following functions satisfies.

- (1) Pu with $P \in D_n$.
- (2) The sum $u + v$.
- (3) The restriction $u|_Y$ of u to an affine subspace of \mathbb{R}^n if it is well-defined as in the case that u is smooth.
- (4) The product uv if it is well-defined as in the case that u is C^∞ and v is a distribution.
- (5) The definite integral $\int_{\mathbb{R}^{n-d}} u(x_1, \dots, x_{n-d}, x_{n-d+1}, \dots, x_n) dx_{n-d+1} \cdots dx_n$ with parameters if it is well-defined as in the case that u has a compact support with respect to the integration variables.

Let us explain briefly how to compute holonomic ideals for functions above. Let I and J be holonomic ideals for u and v respectively. First, a holonomic ideal for Pu can be computed as an ideal quotient $I : P$, and a one for $u + v$ as an ideal intersection $I \cap J$, both by using Gröebner bases.

The restriction algorithm was given in [5] for one codimensional case and in [8] for the general case. For this, one needs a Gröbner base of the ideal I with respect to an ordering compatible with the weight vector of type

$$w = (0, \dots, 0, -1, \dots, -1; 0, \dots, 0, 1, \dots, 1).$$

A holonomic ideal for $u(x)v(y)$ is generated by I and J over D_{2n} . Then by restricting $u(x)v(y)$ to the diagonal $x = y$, we obtain a holonomic ideal. Finally, the definite integral with parameters can be computed also by the restriction algorithm applied to the Fourier transform of I (cf. [7], [9], [6]).

The theory of D -modules assures that the functions obtained by the operations (1)–(5) above are holonomic (See for example, [2]). The outputs of our algorithms are also holonomic since they follow the D -module theoretic procedures precisely, not in a heuristic way as in the pioneering works of Almkvist-Zeilberger [1] and Takayama [10].

4 Definite integrals with the Heaviside function

Let u be a holonomic function on \mathbb{R}^n . Let f_1, \dots, f_m be nonzero polynomials in $x = (x_1, \dots, x_n)$ with real coefficients. Setting

$$D(x_1, \dots, x_d) = \{(x_{d+1}, \dots, x_n) \in \mathbb{R}^{n-d} \mid f_1(x) \geq 0, \dots, f_m(x) \geq 0\},$$

let us consider the definite integral

$$v(x_1, \dots, x_d) = \int_{D(x_1, \dots, x_d)} u(x) dx = \int_{\mathbb{R}^n} Y(f_1) \cdots Y(f_m) u(x) dx.$$

We suppose that this integral is well-defined. This is the case, for example, when u is smooth and $D(x_1, \dots, x_d)$ is compact for any $(x_1, \dots, x_d) \in \mathbb{R}^d$. Then a holonomic ideal for this integral can be computed by combining the algorithms explained so far. If the integrand is the exponential of a polynomial, a power of a polynomial, or the product of such functions, then there are some shortcuts to follow. For practical computations, a library file `nk_restriction.rr` of a computer algebra system Risa/Asir provides one of the most efficient implementations of the restriction and integration algorithms.

Example 3 Consider the definite integral

$$v(t) = \int_{D(t)} \sqrt{x+t} \, dx dy = \int_{\mathbb{R}^2} Y(1-x^2-y^2)(x+t)_+^{\frac{1}{2}} \, dx dy$$

with $D(t) = \{(x, y) \in \mathbb{R}^2 \mid x^2 + y^2 \leq 1, x+t \geq 0\}$. Then by the algorithm described above, we know that $v(t)$ satisfies a linear ordinary differential equation

$$4(1-t^2) \frac{d^2v}{dt^2} + 8t \frac{dv}{dt} - 5v = 0 \quad (3)$$

as a distribution on \mathbb{R} . In fact, $v(t)$ is continuous on \mathbb{R} but may fail to be infinitely differentiable at the singular points $t = \pm 1$ of the equation (3).

References and Notes

- [1] Almkvist, G., Zeilberger, D., The method of differentiating under the integral sign. *J. Symbolic Computation* **10** (1990), 571–591.
- [2] Björk, J.-E., *Rings of Differential Operators*. North-Holland, 1979.
- [3] Nakayama, H., `nk_restriction.rr`: a library file of Risa/Asir, <http://www.math.kobe-u.ac.jp/Asir/asir.html> (2009)
- [4] Oaku, T., Computation of the characteristic variety and the singular locus of a system of differential equations with polynomial coefficients. *Japan J. Indust. Appl. Math.* **11** (1994), 485–497.
- [5] Oaku, T., Algorithms for b -functions, restrictions, and algebraic local cohomology groups of D -modules. *Advances in Appl. Math.* **19** (1997), 61–105.
- [6] Oaku, T., Shiraki, Y., Takayama, N., Algebraic algorithms for D -modules and numerical analysis. *Computer mathematics (Proceedings of ASCM 2003)*, 23–39, Lecture Notes Ser. Comput., 10, World Sci. Publ., River Edge, NJ, 2003.
- [7] Oaku, T., Takayama, N., An algorithm for de Rham cohomology groups of the complement of an affine variety. *J. Pure Appl. Algebra* **139** (1999), 201–233.
- [8] Oaku, T., Takayama, N., Algorithms for D -modules — restriction, tensor product, localization, and local cohomology groups. *J. Pure Appl. Algebra* **156** (2001), 267–308.
- [9] Saito, M., Sturmfels, B., Takayama, N., *Gröbner Deformations of Hypergeometric Differential Equations*. Springer Verlag, 2000.
- [10] Takayama, N., An approach to the zero recognition problem by Buchberger algorithm. *J. Symbolic Computation* **14** (1992), 265–282.

Computational Illusion Toward Escher and Beyond Escher

KOKICHI SUGIHARA

Meiji Institute for Advanced Study of Mathematical Sciences, Meiji University,
1-1-1 Higashimita, Tama-ku, Kawasaki, Kanagawa 214-8571, Japan
kokichis@isc.meiji.ac.jp

.....

Abstract

M. C. Escher, a Dutch woodcut artist, is one of a few artists who utilized mathematical structures in creating artworks. We study three groups of Escher's works, isohedral-tiling art, Sky-and-Water-type art, and pictures of impossible objects, from a mathematical point of view. In particular, we consider how to generate Escher-like art by purely mathematical algorithms, and how to extend them to generalize his art patterns.

Keywords: Illusion, Escher, tiling, impossible objects, figure-ground reversal.

1 Introduction

Mathematics and art are usually very far apart, but come close together in some special cases. M. C. Escher, a Dutch woodcut artist, is a remarkable artist who shortened the distance between mathematics and art by explicitly utilizing mathematical structures. Typical examples are isohedral tiling in “Regular Division of the Plane No. 56 (Lizard)” (1942), hyperbolic-space tiling in “Circle Limit IV” (1960), continuously changing tiling in “Sky and Water I” (1938) and unrealizable motion in “Waterfall” (1961).

Indeed, Escher's artworks have been analyzed from a mathematical point of view in a variety of ways. Escher's tiling artworks were classified according to the mathematical categorization of isohedral tilings [3, 7]. Creation of Escher-like tiling patterns by computers has also been tried for isohedral tiling by Cervini et al. [1] and Kaplan and Salesin [4, 5].

However, simple introduction of mathematical structures to art might just generate abstract patterns. What is remarkable in Escher is that he combined mathematical structures with optical illusions, and thus made mathematical structures nontrivial from an artistic point of view. The visual effects used by Escher include continuous morphing, animal-like complicated tiles, figure-ground reversal, and impossible objects and motions.

Therefore, to generate Escher-like art using computers, we must consider not only geometric structures but also the effects of visual illusion.

We show three examples of trials to combine mathematical structures and visual illusion aiming at computer-aided systems to generate Escher-like art patterns. They are design of complicated isohedral tilings, generation of Sky-and-Water-type tilings, and three-dimensionalization of impossible objects and impossible motions, which are presented in Sections 2, 3 and 4, respectively.

2 Isohedral Tiling and Faithful Escherization

The first group of Escher's works we consider is tiling art with mutually congruent tiles. This group is based on a geometric structure called isohedral tilings.

A partition of the plane into topological disks and their boundaries is called a *tiling*. A tiling is said to be *monohedral* if all the tiles are congruent. A monohedral tiling is said to be *isohedral* if there is a subgroup of the group of congruent transformations in the plane such that the tiling is invariant under those transformations and any pair of tiles has a congruent transformation in the subgroup that maps one to the other.

In isohedral tiling, the relation of one tile with the surrounding tiles around it is the same for every tile. Hence, the same deformation rules can be applied to all the tiles simultaneously and the resulting structures remain a tiling. This way, we can modify the shape of the tiles into a complicated one such as an animal or a human. This is what Escher did in his work notes. In other words, Escher started with a simple initial tiling, and modified the tiles to the shape he wanted.

In computers, on the other hand, we can move in the opposite direction. That is, we first choose an arbitrary goal shape, and next search for a tile that is similar to the goal shape and that admits an isohedral tiling. In this process, visually interesting tilings will be generated if we can find tiles that are close to the goal shape.

Suppose that we are given a figure represented by a cyclic sequence of n points on the boundary of the figure. Let W be a $2 \times n$ matrix composed of the x and y coordinates of the n points. We call W the *goal shape*. Let U be another $2 \times n$ matrix such that:

$$U = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ y_1 & y_2 & \cdots & y_n \end{pmatrix}, \quad (1)$$

by which we want to represent the shape of a tile that admits an isohedral tiling and that is close to W . That is, U is the tile we want to search for; so the entities of U are known variables. We assume that both shapes are placed so that their center of gravity coincides with the origin of the coordinate system.

To measure the distance between U and W , we define the index:

$$D^2(U, W) = \min_{s, \theta} \left\| sR(\theta) \frac{U}{\|U\|} - \frac{W}{\|W\|} \right\|^2, \quad (2)$$

where $\|X\|$ denotes the Frobenius norm of matrix X , s is a scalar representing the scale transformation, and $R(\theta)$ represents the rotation of the figure by angle θ around the origin.

The above distance is convenient because we can rewrite it as:

$$D^2(U, W) = 1 - \frac{\|UW^T\|^2 + 2 \det(UW^T)}{\|U\|^2 \|W\|^2}, \quad (3)$$

and thus we can remove the minimum operation with respect to s and θ [11].

We want to find a shape U that admits an isohedral tiling. Isohedral tilings were classified into 93 types and they are represented by symbols IH01, IH02, ..., IH93 [2]. If we fix the type of the isohedral tiling, we can represent the constraint as a relationship among the points on the boundary of U . For example, suppose that U admits the isohedral tiling of type IH07, which is generated by rotations by $2\pi/3$ around two points. Let p_i, p_j, p_k be three points on the boundary of U such that rotating the plane around p_j by $2\pi/3$ results in p_i moving to the position previously occupied by p_k . Then:

$$R(2\pi/3)(p_i - p_j) = p_k - p_j, \quad (4)$$

which is represented by linear constraints with respect to the x and y coordinates of the three points x_i, y_i, x_j, y_j, x_k and y_k . Collecting similar constraints:

$$A\mathbf{u} = 0, \quad (5)$$

where $\mathbf{u} = (x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_n)^T$ and A is a constant matrix associated with the isohedral tiling type IH07. For other isohedral tilings, we similarly obtain linear constraints depending on the type.

On the other hand, the second term of eq. (3) can be rewritten as:

$$\frac{\|UW^T\|^2 + 2 \det(UW^T)}{\|U\|^2 \|W\|^2} = \frac{\mathbf{u}^T V \mathbf{u}}{\mathbf{u}^T \mathbf{u}}, \quad (6)$$

where V is a symmetric matrix depending on W .

Therefore, if we fix the type of the isohedral tiling, the problem of finding the tile that is similar to the goal shape and that admits isohedral tiling can be reduced to the optimization problem:

$$\begin{aligned} & \text{maximize} && \frac{\mathbf{u}^T V \mathbf{u}}{\mathbf{u}^T \mathbf{u}} \\ & \text{subject to} && A\mathbf{u} = 0, \end{aligned}$$

which can be solved efficiently [6].

Figure 1 shows an example of a tiling obtained by this method: (a) shows a pair of a goal shape (the larger shape) and the tile found by our method (the smaller shape), and (b) shows the resulting tiling.

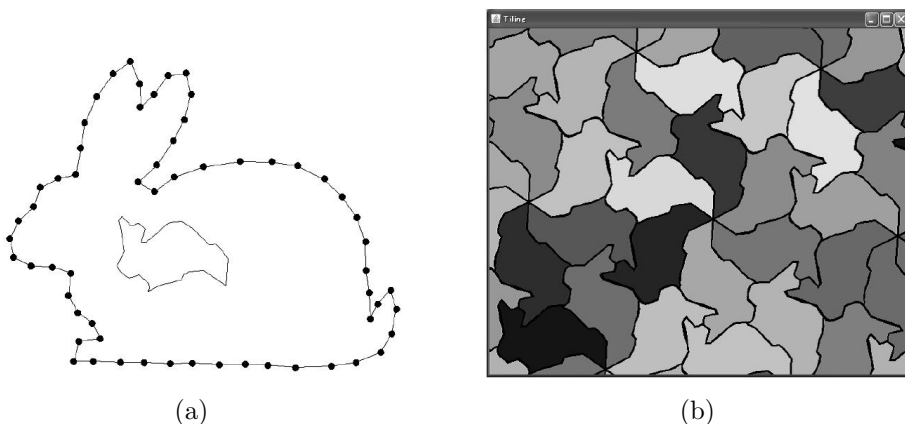


Figure 1: Isohedral tiling generated from a goal shape of a rabbit.

3 Morphing for Sky-and-Water Tilings

The second group of Escher's works we consider is the Sky-and-Water- type tiling, in which a figure at the top changes its shape gradually downward and melts away into the background, and another shape gradually appears from the background. This group of artworks is a combination of tiling, morphing, and figure-ground reversal.

Escher constructed this type of tiling by first generating a dihedral tiling at the intermediate level, and deforming it so that one tile gradually becomes a clear object upward, and the other tile gradually becomes the other clear object downward [8].

In computers, on the other hand, we first fix two goal shapes at the top and at the bottom, and generate intermediate tiles to form the Sky-and-Water tiling pattern. For this purpose, the morphing is applied to the top figure and to the shape of the vacant space surrounded by four copies of the bottom figure [10].

Figure 2 shows an example of the tiling generated in this way: (a) shows a pair of top and bottom figures, and (b) shows the computer-generated tiling.

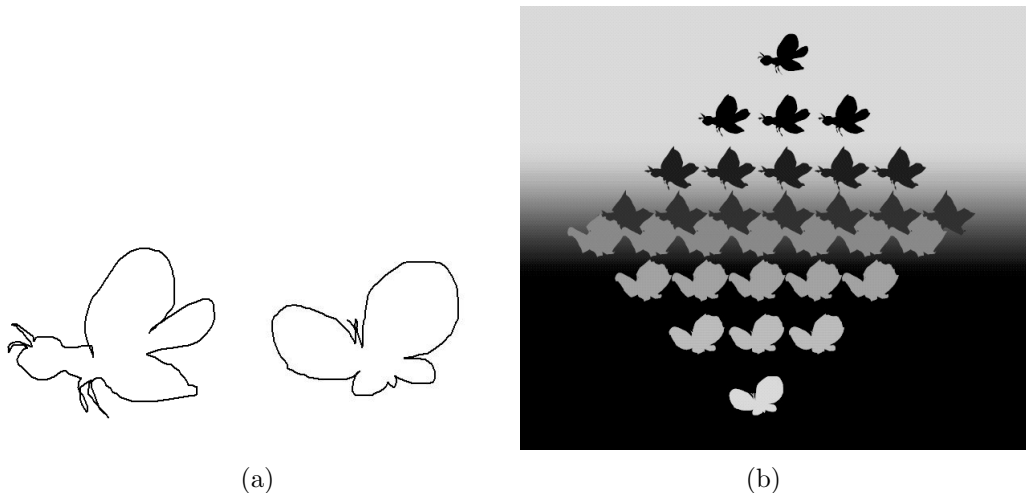


Figure 2: Sky-and-Water tiling generated from a bee and a butterfly.

4 Realization of Impossible Objects and Impossible Motions

The third group of Escher’s works is the class of pictures using impossible objects. A typical example is “Waterfall” (1961), in which water that falls down runs through a water path upward to the start point of the fall again, and thus the water runs in a cyclic manner forever. This is physically impossible because it contradicts the impossibility of an eternal engine.

Escher drew his impossible objects and impossible motions only on a plane. On the other hand, we found that some of those pictures are realizable as actual objects and actual motions in the three-dimensional world through optical illusion.

The picture lacks depth information, and hence there is great freedom in reconstructing the three-dimensional objects from the picture. Therefore, even if a picture looks impossible, we can sometimes construct a solid, although the resulting solid is quite different from our intuition. With the same trick, we can also create impossible-looking motions. Such impossible objects and motions can be searched for by a computer in the following manner.

Suppose that the viewpoint is fixed at the origin of the xyz Cartesian coordinate system, and the picture is fixed on the plane $z = 1$. Let $v_i = (x_i, y_i, 1)$ be the i th vertex in the picture, and let $(x_i/t_i, y_i/t_i, 1/t_i)$ be the associated vertex on the three-dimensional solid whose projection coincides with the given picture, where t_i is an unknown variable

representing the depth of v_i . Furthermore, let the j th face of the solid be represented by the equation:

$$a_j x + b_j y + c_j z + 1 = 0. \quad (7)$$

Let m be the number of vertices and n be the number of faces represented in the picture.

Suppose that the i th vertex is on the j th face. Then:

$$a_j x_i + b_j y_i + c_j + t_i = 0. \quad (8)$$

For all pairs of vertices and faces containing them, we obtain similar equations, and thus we can construct a system of linear equations, which we represent by:

$$A\mathbf{w} = 0, \quad (9)$$

where \mathbf{w} is a vector of unknown parameters:

$$\mathbf{w} = (t_1, t_2, \dots, t_m, a_1, b_1, c_1, \dots, a_n, b_n, c_n)^T, \quad (10)$$

and A is a constant matrix.

Next, suppose that the k th vertex is behind the l th face when it is extended. Then:

$$a_l x_k + b_l y_k + c_l + t_k < 0. \quad (11)$$

Collecting all such inequalities yields a system of linear inequalities:

$$B\mathbf{w} > 0. \quad (12)$$

The set of all solids that generate a given projected picture is represented by the feasible region specified by the linear equations (9) and inequalities (12).

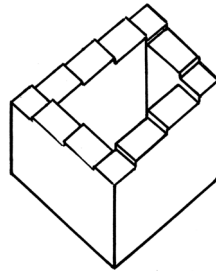
This feasible region is fragile in the sense that it may become empty when even a slight numerical error occurs in the plane. However, we constructed a method to snap the vertex positions in the picture to the correct locations, and thus established a robust method to judge the realizability of solids from a picture [9]. Using this method, we can construct solids for some pictures of “impossible objects” and can also generate impossible motions using those solids.

Figure 3 shows an example of a realization of an impossible object: (a) shows a picture of an impossible object, (b) shows a realization of the associated solid, and (c) shows the same solid seen from another angle.

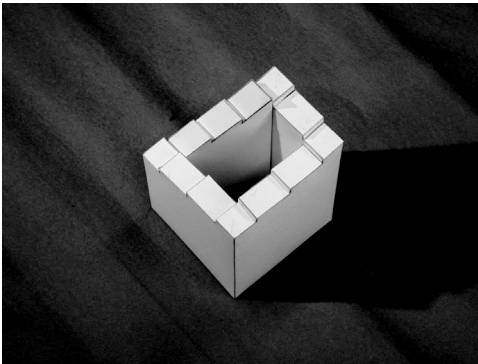
5 Concluding Remarks

We have presented three examples of computer-aided methods, by which we can generate Escher-like tilings and three-dimensional solids that realize Escher-like impossible objects and motions. The success of these computer-aided methods means that not only can we realize the mathematical structures behind Escher’s works, we can also realize visual effects by applying computer power to the search for the optimal shapes.

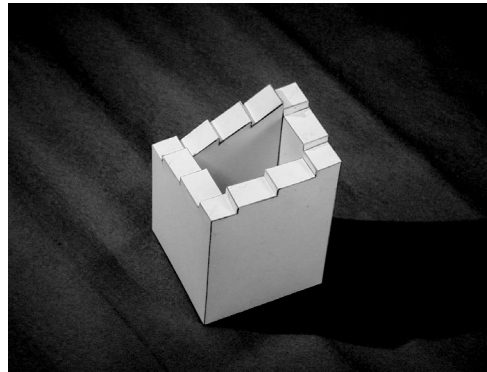
Escher also generated many other types of artwork, most of which are also based on mathematical structures. Hence, there still remain many possibilities for utilizing computer power to generate those artistic patterns automatically and effectively.



(a)



(b)



(c)

Figure 3: Realization of a solid that appears to be an impossible object.

Acknowledgments

The computer-aided isohedral tiling presented in Section 2 is a joint work with Mr. H. Koizumi, a Master's student of the Department of Mathematical Informatics of the University of Tokyo. This work is supported by the Grant-in-Aid for Scientific Research (B) No. 20360044 and the Grant-in-Aid for Exploratory Research No. 19650003 of the Japanese Society for Promotion of Science.

References and Notes

- [1] L. Cervini, R. Farinato and L. Loreto: The interactive computer graphics (ICG) production of the 17 two-dimensional crystallographic groups, and other related topics. In H. S. M. Coxeter, M. Emmer, R. Penrose and M. L. Teuber (eds.): *M. C. Escher — Art and Science*, pp. 269–284, North-Holland, Amsterdam, 1986.
- [2] B. Grünbaum and G. C. Shephard: *Tiling and Patterns — An Introduction*. W. H. Freeman, New York, 1989.
- [3] C. S. Kaplan: Metamorphosis in Escher's art. In *Bridges 2008 — Mathematical Connections in Art, Music and Science*, July 24–28, Leeuwarden, the Netherlands, pp. 39–46, 2008.

Invited talks

- [4] C. S. Kaplan and D. H. Salesin: Escherization. *Proceedings of SIGGRAPH 2000*, New Orleans, LA, pp. 499–510, 2000.
- [5] C. S. Kaplan and D. H. Salesin: Dihedral Escherization. *Proceedings of Graphics Interface 2004*, May 2004, London, Ontario, pp. 255–262, 2004.
- [6] H. Koizumi and K. Sugihara: Computer aided design system for Escher-like tilings. *Proceedings of the 25th European Workshop on Computational Geometry*, March 2009, Brussels, pp. 345–348, 2009.
- [7] D. Schattschneider: *M. C. Escher — Visions and Symmetry*, New Edition. Abrams, New York, 2004.
- [8] D. Schattschneider: Lessons in duality and symmetry from M. C. Escher. In *Bridges 2008 — Mathematical Connections in Art, Music and Science*, July 24–28, Leeuwarden, the Netherlands, pp. 1–8, 2008.
- [9] K. Sugihara: Computer-aided creation of impossible objects and impossible motions. *The Kyoto International Conference on Computational Geometry and Graph Theory*, Kyoto, June 2007.
- [10] K. Sugihara: Computer-aided generation of Escher-like Sky and Water tiling patterns. *Journal of Mathematics and the Art* (to appear).
- [11] M. Werman and D. Weinshall: Similarity and affine invariant distances between 2D point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, pp. 810–814, 1995.

A Symbolic-numeric Algorithm for Computing the Multiple Roots of Polynomial Systems Accurately

LIHONG ZHI¹

¹ Key Lab of Mathematics Mechanization, AMSS,
Beijing 100190, China
lzhi@mmrc.iss.ac.cn

Extended Abstract

Consider an ideal I generated by a polynomial system $F = \{f_1, \dots, f_t\}$, where $f_i \in \mathbb{C}[x_1, \dots, x_s]$, $i = 1, \dots, t$. For a given isolated singular solution $\hat{\mathbf{x}} = (\hat{x}_1, \dots, \hat{x}_s)$ of F , suppose Q is the isolated primary component whose associate prime is $P = (x_1 - \hat{x}_1, \dots, x_s - \hat{x}_s)$, in [Wu and Zhi, 2008a], we use symbolic-numeric method based on the geometric jet theory of partial differential equations introduced in [Bonasia et al., 2004, Reid et al., 2003, Zhi and Reid, 2004] to compute the index ρ and multiplicity μ , such that $Q = (I, P^\rho)$ and $\mu = \dim(\mathbb{C}[\mathbf{x}]/Q)$. We also derive a simple involutive criterion based on the special structure of the ideal (I, P^k) and apply it to the truncated coefficient matrices formulated from the Taylor series expansions of polynomials in prolonged systems of F at $\hat{\mathbf{x}}$ to order k . The number of columns of these coefficient matrices is fixed by $\binom{k+s-1}{s}$. A basis for the Max Noether space of I at $\hat{\mathbf{x}}$ is obtained from the null space of the truncated coefficient matrix of the involutive system.

If a singular solution is only known with limited accuracy, by choosing a tolerance, we can compute the index, multiplicity and a basis of the Max Noether space for this approximate singular solution. It is well known that numeric computations deeply depend on the choice of tolerance. In order to obtain accurate information about the multiplicity structure, we present in [Wu and Zhi, 2008a] an algorithm to improve the accuracy of the singular root. Suppose $\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{exact}} + \hat{\mathbf{x}}_{\text{error}}$, where $\hat{\mathbf{x}}_{\text{exact}}$ denotes the exact singular solution of F and $\hat{\mathbf{x}}_{\text{error}}$ denotes the error in the solution, we compute the truncated coefficient matrix of the involutive system by shifting the coefficient matrix formulated from the truncated multivariate Taylor series expansions of polynomials in F at $\hat{\mathbf{x}}$ to order ρ , then generate multiplication matrices from its null vectors. Let $\hat{\mathbf{y}}$ be the averages of the traces of the multiplication matrices. In [Wu and Zhi, 2008b], we prove that if the given singular solution satisfies $\|\hat{\mathbf{x}} - \hat{\mathbf{x}}_{\text{exact}}\| = \mathcal{O}(\varepsilon)$, and the index and multiplicity of the singular solution are computed correctly, then the refined solution obtained by adding $\hat{\mathbf{y}}$ to $\hat{\mathbf{x}}$ will satisfy $\|\hat{\mathbf{x}} + \hat{\mathbf{y}} - \hat{\mathbf{x}}_{\text{exact}}\| = \mathcal{O}(\varepsilon^2)$. If we underestimate or overestimate the index due to poorly chosen tolerance, then we can rediscover the correct index after the accuracy of the approximate singular solution improved after one or two iterations.

The size of these coefficient matrices in [Wu and Zhi, 2008a,b] is bounded by $t \binom{\rho+s-1}{s} \times \binom{\rho+s-1}{s}$ which will be very big when ρ or s is large. In general $\rho \leq \mu$, however, when the corank of the Jacobian matrix $J(\hat{\mathbf{x}})$ is one, then $\rho = \mu$, which is also called the breadth one case in [Dayton et al., 2009, Dayton and Zeng, 2005], the sizes of the matrices can easily exceed the storage capacity of a personal computer. This is the main motivation for us to consider whether we can compute the multiplicity structure of $\hat{\mathbf{x}}$ efficiently in this worst

*This is the joint work with my Ph.D students Xiaoli Wu and Nan Li.

†Supported by NKBRPC (2004CB318000) and the Chinese National Natural Science Foundation under Grant 10401035 and 60821002/F02.

case. In Li and Zhi [2009], we prove in the breadth one case, the number of free parameters used in computing each order of the Max Noether condition of I at $\hat{\mathbf{x}}$ can be reduced to $s-1$. Therefore, in order to determine a closed basis of the Max Noether space incrementally for k from 2 to $\mu-1$, we only need to check whether a computed vector \mathbf{p}_k can be written as a linear combination of the last $s-1$ linear independent columns of the Jacobian matrix $J(\hat{\mathbf{x}})$. If \mathbf{p}_k is not consistent with $J(\hat{\mathbf{x}})$, then we are finished. Otherwise, the coefficients of the linear combination will give us unique values of those free parameters and we find a new Max Noether condition of the k -th order. The size of matrices we used in computing each order of the Max Noether conditions is bounded by $t \times (s-1)$. It does not depend on the multiplicity. Moreover, during the computation, we only need to store polynomials, the LU decomposition of the last $s-1$ columns of the Jacobian matrix and the computed Max Noether conditions. Therefore, in the breadth one case, both storage space and execution time for computing a closed basis of the Max Noether space are reduced significantly. We also apply these strategies to reduce the matrices appeared in [Wu and Zhi, 2008a] to obtain a more efficient algorithm for refining an approximately known multiple root for this special case.

References and Notes

- Bonasia, J., Lemaire, F., Reid, G., Scott, R., and Zhi, L. Determination of approximate symmetries of differential equations. In *CRM Proceedings and Lecture Notes*, pages 233–249, 2004.
- Dayton, B., Li, T., and Zeng, Z. Multiple zeros of nonlinear systems. <http://orion.neiu.edu/~zzeng/Papers/MultipleZeros.pdf>, 2009.
- Dayton, B. and Zeng, Z. Computing the multiplicity structure in solving polynomial systems. In *ISSAC'05 Proc. 2005 Internat. Symp. Symbolic Algebraic Comput.*, pages 116–123, 2005.
- Li, N. and Zhi, L. Compute the multiplicity structure of an isolated singular solution: Case of breadth one. Manuscript, 2009.
- Reid, G., Tang, J., and Zhi, L. A complete symbolic-numeric linear method for camera pose determination. In Sendra, J., editor, *Proc. 2003 Internat. Symp. Symbolic Algebraic Comput. ISSAC'03*, pages 215–223, New York, 2003. ACM Press. ISBN 1-58113-641-2.
- Wu, X. and Zhi, L. Computing the multiplicity structure from geometric involutive form. In *Proc. 2008 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'08)*, pages 325–332, 2008a.
- Wu, X. and Zhi, L. Determining singular solutions of polynomial systems via symbolic-numeric reduction to geometric involutive form. *MM Research Preprints*, 27:104–122, 2008b.
- Zhi, L. and Reid, G. Solving nonlinear polynomial system via symbolic-numeric elimination method. In Faugère, J. and Rouillier, F., editors, *Proc. International conference on polynomial system solving*, pages 50–53, 2004. Full version of the paper in *J. Symbolic Comput.* 44(3), 280–291.