Journal of Energy Technology

JET Volume 8 (2015) p.p. 43-58
Issue 2, October 2015
Typology of article 1.01
www.fe.um.si/en/jet.html

INTEGER PROGRAMMING AND GRÖBNER BASES

CELOŠTEVILSKO PROGRAMIRANJE IN GRÖBNERJEVE BAZE

Brigita Ferčec[1], Matej Mencinger[R]

Keywords: Integer linear programming, polynomial rings, Gröbner bases, nonlinear polynomial systems of equations

Abstract

An approach to solve the integer linear programming problem (IP) using the Gröbner bases theory is presented. We consider the basics of commutative algebra on polynomial rings and their ideals and the multidivision algorithm. Gröbner bases were introduced to solve nonlinear polynomial systems of equations; therefore, we first present the generalization of the Gauss elimination method. In order to solve a general IP a special ideal depending on the coefficients of the system and number of constraints in the IP has to be constructed. Finally, a Gröbner basis of this ideal, which yields the solution to IP, must be sought.

Povzetek

V članku je podan pristop k reševanju celoštevilskega linearnega programiranja (IP) z uporabo teorije Gröbnerjevih baz. Obravnavamo osnovne elemente komutativne algebre na polinomskih kolobarjih, njihove ideale in algoritem multi-deljenja. Gröbnerjeve baze so bile vpeljane za reševanje nelinearnih polinomskih sistemov enačb, zato je v članku najprej predstavljen primer posplošitve Gaussove eliminacijske metode. Pri reševanju splošnega IP konstruiramo poseben ideal, ki je odvisen od koeficientov sistema in števila enačb v IP. Končno rešitev dobimo s pomočjo Gröbnerjeve baze tega ideala.

[R] Corresponding author: Integer programming and Gröbner bases, Matej Mencinger, Tel.: +386 2 229 4321, Mailing address: University of Maribor, Faculty of Civil Engineering, Transportation Engineering and Architecture, Smetanova ulica 17, 2000 Maribor, Slovenia, E-mail address: matej.mencinger@um.si

[1]Faculty of Energy Technology, Hočevarjev trg 1, 8270 Krško

1 INTRODUCTION

Many modern computer programs, such as Mathematica, Matlab and others, enable solving the problem of integer (linear) programming (IP). There are several algorithms to solve the problem of IP; one of the most known and commonly used is the so called 'Branch and bound algorithm'. In this paper, we consider another aspect of solving this problem, which is based on the theory of Gröbner bases, which is the basis of the ideal of a polynomial ring in a similar sense as the vector space $(\mathbb{R}^n, +)$, which has a basis consisting of n linearly independent vectors (1,0,0,…,0), (0,1,0,…,0), ..., (0,0,0,…,1). These n vectors are called a standard basis of $(\mathbb{R}^n, +)$ and in a similar sense theGröbner basis is called a standard basis of the given ideal.

Turning now to the IP, where there are more equations and variables, and a cost function to be optimized, and taking into account that the main objects of the polynomials of several variables are monomials, we can use the fact that the exponent of each monomial is actually a vector. As we will see later, this vector is naturally related to independent variables of IP that are (basically positive) integers.

There are many different ways of examining the theory of Gröbner bases. In the context of classical algebra, the natural point of view is as follows.

We consider polynomials in variables $x_1, \dots, x_n$ with coefficients a_α of a field k. We call $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ a *monomial*, $a_\alpha \in k$ a *coefficient* and $a_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ a *term*, and $\alpha = (\alpha_1, \dots, \alpha_n)$ a *multi-index*. The set of all polynomials in variables $x_1, \dots, x_n$ with coefficient in k is denoted by $k[x_1, \dots, x_n]$. With the usual operations of addition and multiplication, $k[x_1, \dots, x_n]$ is a commutative ring (i.e. the multiplication in the ring is symmetric). We call $\alpha = \alpha_1 + \cdots + \alpha_n$ the full degree of a monomial x^α. The degree of a polynomial f, denoted by $\deg(f)$, is the maximum degree of a monomial of f. For any natural number n, the space

$$k^n = \{(a_1, \dots, a_n) : a_1, \dots, a_n \in k\}$$

is called n −dimensional *affine space*. The set of polynomials $f_1, \dots, f_s$ is naturally associated with a system of equations

$$\begin{aligned} f_1(x_1, \dots, x_n) &= 0, \\ &\vdots \\ f_s(x_1, \dots, x_n) &= 0. \end{aligned} \tag{1.1}$$

The set of all solutions to the above system can be defined as the *affine variety*, $\boldsymbol{V}$, determined by polymomials $f_1, \dots, f_s$:

$$\boldsymbol{V}(f_1, \dots, f_s) = \{(a_1, \dots, a_n) \in k^n : f_j(a_1, \dots, a_n) = 0 \; for \; 1 \le j \le s\}.$$

It is clear that there are many families of polynomials defining the same variety. For example, if $f_1 = x - y$, $f_2 = y - 1$ and $g_1 = (y - x)^3$, $g_2 = y^2 - 2y + 1$, then $\boldsymbol{V}\langle f_1, f_2\rangle = \boldsymbol{V}\langle g_1, g_2\rangle$. To understand the concept of an affine variety better we need the notion of an ideal.

An *ideal* in the polynomial ring $k[x_1, \dots, x_n]$ is a subset I of $k[x_1, \dots, x_n]$ satisfying

i. if $f, g \in I$ then $f + g \in I$ and
ii. if $f \in I$ and $h \in k[x_1, \dots, x_n]$, then $hf \in I$.

Let $f_1, \dots, f_s$ be polynomials of $k[x_1, \dots, x_n]$. We denote

$$\langle f_1, \dots, f_s \rangle = \left\{ \sum_{j=1}^{s} h_j f_j \mid h_1, \dots, h_s \in k[x_1, \dots, x_n] \right\}.$$

It is easily seen that $\langle f_1, \dots, f_s \rangle$ is an ideal in $k[x_1, \dots, x_n]$. Polynomials $f_1, \dots, f_s$ are called *generators* of this ideal. Ideal $I \subset k[x_1, \dots, x_n]$ is *finitely generated* if there exist polynomials $\langle f_1, \dots, f_s \rangle \in k[x_1, \dots, x_n]$ such that $I = \langle f_1, \dots, f_s \rangle$, and the set $\{f_1, \dots, f_s\}$ is called a *basis* of I.

From the definition of an affine variety, we see that to find an affine variety $\boldsymbol{V}(f_1, \dots, f_s) \subset k^n$ is equivalently as to find the set of solutions of system (1.1). This is a problem that frequently arises in studies of various phenomena in physical, technical and other sciences. In particular, in order to study the qualitative behaviour of dynamics system

$$\dot{x}_1 = f_1(x_1, x_2, \dots, x_n) = 0, \dots, \dot{x}_s = f_s(x_1, x_2, \dots, x_n)$$

we first have to determine singular points of the system, which are the points where all polynomials $f_1, \dots, f_s$ vanish. Thus, we again arrive at a problem of the form (1.1), [6,7,10].

The problem of finding solutions of system (1.1) is very difficult. It can happen that system (1.1) has infinitely many solutions, which means that it is impossible to find (all) solutions numerically. Even if system (1.1) has a finite number of solutions it is still very difficult and often impossible to find all of them numerically without applying methods of computational algebra.

In the next section, we describe the concept of Gröbner bases, which is closely related to the multivariable division algorithm. We consider the main properties of Gröbner bases and connect them to the solution of system (1.1). In the third section, we study the main topic of this paper, the integer programming problem (IP) via the theory of Gröbner bases. We demonstrate the application of the theory of Gröbner bases theory to IP on two examples. In the first example, we consider a case of IP without a cost function (i.e. just a problem to find integer solutions to $A\vec{x} = \vec{b}$), while in the second example we search for the solutions to a general IP (with a given cost function).

2 GRÖBNER BASIS AND NONLINEAR SYSTEMS OF EQUATIONS

Until the mid-1960s, when Bruno Buchberger, [2], invented the theory of Gröbner bases, no method for solving a general (nonlinear polynomial) system (1.1) was known. What is nowadays called Buchberger's algorithm (and Gröbner basis) is actually the cornerstone of modern computational algebra. In this section, we first briefly describe the notion of a Gröbner basis, which will be used to obtain the variety of an ideal generated by polynomials $f_1, \dots, f_s$, i.e. to obtain the solution of system (1.1). Since a Gröbner basis of I depends on a term ordering on monomials of $k[x_1, \dots, x_n]$, we define monomial (term) order.

A *monomial order* $<$ on $k[x_1, \dots, x_n]$ is a total order $<$ on $\mathbb{N}_0^n$ with the following two properties:

(i) any nonempty subset of monomials has a least element (under <),

(ii) if $x^\alpha < x^\beta$ then $x^\alpha x^\gamma < x^\beta x^\gamma$ for every monomial x^γ.

The most common monomial ordering is shown in the following example.

Let us consider $x_1^2x_2^8x_3^{50}, x_1^3x_2^2x_3^5, x_1^2x_2^9x_3^4 \in \mathbb{R}[x_1, x_2, x_3]$ and say, x_1 is 'more important' than x_2 (and x_3) and x_2 is 'more important' than x_3. Then:

$$x_1^3x_2^2x_3^5 > x_1^2x_2^9x_3^4 > x_1^2x_2^8x_3^{50}.$$

This monomial order is called a *lexicographic (monomial) order*. More precisely, $x^\alpha < x^\beta$ if and only if the first coordinates $\alpha_i$and β_i in α and β from left, which are different, satisfy $\alpha_i < \beta_i$. There are many other standard monomial orders, like (graded) reverse lexicographic, (graded) reverse lexicographic, etc. [4]. To avoid the, we can emphasize the name of the monomial order. For example $<_{lex}$ stands for the lexicographic order.

As soon as the monomial order is chosen, we can speak of the so-called *leading monomial* (LM), *leading term* (LT) and *leading coefficient* (LC) of the polynomial. The leading term is defined with the largest (with respect to fixed order) monomial.

For example, if $f = -x_1^2x_2^8x_3^{50} + 2x_1^3x_2^2x_3^5 + 3x_1^2x_2^9x_3^4$ and the order is lexicographic with $x_1 > x_2 > x_3$, the leading term of f is $LT(f) = 2x_1^3x_2^2x_3^5$, whilst the leading coefficient is $LC(f) = 2$ and the leading monomial $LM(f) = x_1^3x_2^2x_3^5$.

Finally, note that any vector $\vec{c} \in \mathbb{R}^n$ defines an monomial order $<_{\vec{c}}$ in $\mathbb{R}[x_1, \dots, x_n]$ in the following way:

$$\vec{x}^{\vec{\alpha}} <_{\vec{c}} \vec{x}^{\vec{\beta}} \Leftrightarrow \begin{cases} \vec{c} \cdot \vec{\alpha} < \vec{c} \cdot \vec{\beta} & or \\ \vec{c} \cdot \vec{\alpha} = \vec{c} \cdot \vec{\beta} \quad and\ \vec{\alpha} <_{lex} \vec{\beta}, \end{cases}$$

where $\vec{c} \cdot \vec{\alpha}$ stands for the standard dot product $\vec{c} \cdot \vec{\alpha} = \sum_i c_i\alpha_i$. This monomial term order $<_{\vec{c}}$ defined by vector $\vec{c}$ is usually called *weighted* (monomial) *order*.

For example, if $\vec{c} = (1,5,10)$, we have $x_1^5x_2^1x_3^2 <_{\vec{c}} x_1^1x_2^0x_3^3$ since $\vec{c} \cdot \vec{\alpha} = (1,5,10) \cdot (5,1,2) = 30$ and $\vec{c} \cdot \vec{\beta} = (1,5,10) \cdot (1,0,3) = 31$ (and $30 < 31$). Note, that for $<_{\vec{c}}$ the leading term of $g = 2x_1^5x_2^1x_3^2 - 5x_1^1x_2^0x_3^3$ is $LT(g) = -5x_1^1x_3^3$ while for $<_{lex}$ the leading term of (the same) g is $LT(g) = 2x_1^5x_2^1x_3^2$.

Now, we describe the procedure of multi-division of a polynomial by an ordered set of polynomials, that is to divide $f \in k[x_1, \dots, x_n]$ by an ordered set $F = \{f_1, \dots, f_s\}$, which means to express f in the form

$$f = q_1f_1 + q_2f_2 + \cdots + q_sf_s + r, \tag{1.2}$$

where the quotients $q_1, \dots, q_s$ and the reminder r are polynomials of $k[x_1, \dots, x_n]$, and either $r = 0$ or $\deg(r) < \deg(f)$. In this case (1.2) means that f is reduced to r modulo F, and we write

$$f \xrightarrow{F} r.$$

The process of multi-division depends on the monomial order and the order of the polynomials in the set $F = \{f_1, \dots, f_s,\}$. So, we have to fix the monomial order first to perform the multidivision. For the precise introduction to multivariable division algorithm see e.g. Adams & Luostaunau (1994), Cox et al. (2007), and Romanovski & Shafer (2009). We demonstrate the division process by the following example.

Let us divide $f = x^2y + xy^2 + y^2$ by the ordered set of polynomials $F = \{f_1, f_2\} = \{xy - 1, y^2 - 1\}$ using lexicographic order with $x > y$.

$$\begin{array}{rl|l}
q_1: & x+y & \\
q_2: & 1 & r \\
f_1: xy-1 & \sqrt{x^2y+xy^2+y^2} & \\
f_2: y^2-1 & & \\
 & \underline{x^2y - x} & \\
 & \quad xy^2+x+y^2 & \\
 & \quad \underline{xy^2 - y} & \\
 & \quad\quad x+y^2+y & \\
 & \quad\quad \underline{y^2+y} \quad \to \quad x & \\
 & \quad\quad\quad y^2-1 & \\
 & \quad\quad\quad \underline{y+1} & \\
 & \quad\quad\quad\quad \underline{1} \quad \to \quad x+y & \\
 & \quad\quad\quad\quad 0 \quad \to \quad x+y+1 &
\end{array}$$

Figure 1: *The scheme of multivariable division procedure*

On the first step, the leading term $LT(f_1) = xy$ divides the leading term $LT(f) = x^2y$. Thus, we divide x^2y by xy, leaving x and then subtract $x \cdot f_1$ from f. Next, we repeat the same process on $x\,y^2 + x + y^2$. We divide f by $LT(f_1) = xy$ again. Note that neither $LT(f_1) = xy$ nor $LT(f_2) = y^2$ divides $LT(x + y^2 + y) = x$. However, $x + y^2 + y$ is not the remainder since $LT(f_2)$ divides y^2. Thus, if we move x to the remainder column, we can continue with the process. If we can divide by $LT(f_1)$ or $LT(f_2)$, we proceed as usual, otherwise, we move the leading term of the intermediate dividend (the polynomial under the radical sign) to the remainder column. We continue dividing in such a way. Now the polynomial under the radical is $y^2 + y$. It is not divisible by $LT(f_1)$ but we can divide it by $LT(f_2)$ yielding 1 and the subtract $1 \cdot y^2$ from $y^2 + y$. The obtained polynomial under the radical is $y + 1$ and neither $LT(f_1)$ nor $LT(f_2)$ is divisible by $LT(y + 1) = y$. Therefore, we move y to the remainder column and obtain 1, which is also moved to the remainder column. Thus, the remainder is $x + y + 1$ and this concludes the example. Thus, we can write f in the form

$$f = x^2y + xy^2 + y^2 = (x + y) \cdot (xy - 1) + 1 \cdot (y^2 - 1) + x + y + 1.$$

In contrast, if we change the order of polynomials f_1 and f_2 in F, i.e. if we divide f by the ordered set $F = \{f_2, f_1\}$, we obtain

$$f = x^2y + xy^2 + y^2 = x \cdot (xy - 1) + (x + 1) \cdot (y^2 - 1) + 2x + 1.$$

Obviously, this multivariable division is very sensitive on the order of f_1, f_2. The order affects the *multi-quotients* q_1, q_2, as well as the *remainder* r. Dividing the polynomial f with the (ordered) set $F = \{f_1, f_2\}$, one can simply write: $f = \{\{q_1, q_2\}, r\}$ instead of $f = q_1f_1 + q_2f_2 + r$. Using this notation, in the first case, we have $f = \{\{x + y, 1\}, x + y + 1\}$ and in the second case we have $f = \{\{x, x + 1\}, 2x + 1\}$. Figure 2 shows the last results obtained by the MATHEMATICA computer algebra system. The multi-quotients and the remainder are also changed if we use another monomial order.

```
In[1]:= PolynomialReduce[x^2 y + x y^2 + y^2, {x y - 1, y^2 - 1}, {x, y}]

Out[1]= {{x + y, 1}, 1 + x + y}

In[2]:= PolynomialReduce[x^2 y + x y^2 + y^2, {y^2 - 1, x y - 1}, {x, y}]

Out[2]= {{1 + x, x}, 1 + 2 x}
```

Figure 2: *Results obtained by MATHEMATICA for the case above*

Now, we present the basic definitions and properties of Gröbner bases. For any ideal, $\boldsymbol{I}$ we define $\langle \boldsymbol{LT(I)} \rangle = \langle \boldsymbol{LT(f)}: \boldsymbol{f} \in \boldsymbol{I} \setminus \{\boldsymbol{0}\} \rangle = \langle \boldsymbol{LM(f)}: \boldsymbol{f} \in \boldsymbol{I} \setminus \{\boldsymbol{0}\} \rangle$. A Gröbner basis of an ideal $\boldsymbol{I} \subset \boldsymbol{k}[\boldsymbol{x_1}, , \dots, \boldsymbol{x_n}]$ is a finite subset $\boldsymbol{G} = \{\boldsymbol{g_1}, \dots, \boldsymbol{g_t}\}$ of $\boldsymbol{I}$ such that

$$\langle LT(I) \rangle = \langle LT(g_1), \dots, LT(g_t) \rangle.$$

It is a special generating set for ideal $\langle f_1, \dots, f_n \rangle$ for which the multivariable division algorithm for a given f returns the remainder $r = 0$ if and only if $f \in \langle g_1, \dots, g_t \rangle$,[4].
Using a Gröbner basis, we obtain the uniqueness of the remainder, which was not assured when we divided by an arbitrary set of polynomials, [2].
We now describe an algorithm for computing a Gröbner basis of a polynomial ring. Let f, g be from $k[x_1, \dots, x_n]$ with $LT(f) = ax^\alpha$ and $LT(g) = bx^\beta$. The *least common multiple* of x^α and x^β, denoted $LCM(x^\alpha, x^\beta)$, is the monomial $x^\gamma = x_1^{\gamma_1} \cdots x_n^{\gamma_n}$ such that $\gamma_j = \max(\alpha_j, \beta_j)$, $1 \leq j \leq n$. The so-called S −polynomial of f and g is

$$S_{f,g} = \frac{x^\gamma}{LT(f)} \cdot f - \frac{x^\gamma}{LT(g)} \cdot g.$$

Buchcberger's basic observation was the following criterion, [2]. Let I be a nonzero ideal in $k[x_1, \dots, x_n]$ and let $<$ be a fixed monomial order on $k[x_1, \dots, x_n]$. Then, $G = \{g_1, g_2, \dots, g_t\}$ is a Gröbner basis for I with respect to $<$ if and only if for all $i \neq j$

$$S_{g_i,\ g_j} \xrightarrow{G} 0.$$

This criterion is the essence of the famous *Buchberger's algorithm*, which produces a Gröbner basis for the nonzero ideal $I = \langle f_1, \dots, f_s \rangle$. Buchberger's algorithm is shown below, [11].

Buchberger's Algorithm

Input: A set of polynomials $\{f_1, \dots, f_s\} \in k[x_1, \dots, x_n] \backslash \{0\}$.
Output: A Gröbner basis G of the ideal $\langle f_1, \dots, f_s \rangle$.
Procedure: $G := \{f_1, \dots, f_s\}$.
Step 1. For each pair $g_i, g_j \in G, i \neq j$, compute the S −polynomial S_{g_i,g_j} and compute the remainder $r_{i,j}$ of division S_{g_i,g_j} by the set G.
Step 2. Check if all $r_{i,j}$ are equal to zero . If "yes", then G is a Gröbner basis, otherwise add all nonzero $r_{i,j}$ to G and return to step 1.

It is proved in [2] that the algorithm terminates and returns a Gröbner basis of the ideal $I = \langle f_1, f_2, \dots, f_n \rangle$.
Nowadays, all well-known computer algebra systems (MATHEMATICA, SINGULAR, MAPLE, REDUCE, and others) have routines to compute Gröbner bases.
Even if a monomial order is fixed, an imprecision in the computation of a Gröbner basis arises because the division algorithm can produce different remainders for different orderings of polynomials in the set of divisors. Thus the output of Buchberger's Algorithms is not unique; neither is it minimal (in the sense that it contains more polynomials then necessary).
A Gröbner basis $G = \{g_1, \dots, g_m\}$ is called minimal if, for all $i, j \in \{1, \dots, m\}$, $LC(g_i) = 1$ and for $j \neq i$, $LM(g_i)$ does not divide $LM(g_j)$. Every nonzero polynomial ideal has a minimal Gröbner basis, [4]. If no term of g_i is divisible by any $LT(g_j)$ for $j \neq i$ then the Gröbner basis is called reduced and if we fix a monomial order then every nonzero ideal $I \subset k[x_1, \dots, x_n]$ has a unique reduced Gröbner basis with respect to this order.

In Figures 3 and 4, the Gröbner basis of the ideal $\langle -x^3 + y, x^2 y - y^2 \rangle$ with respect to lexicographic monomial order with $x > y$ is computed in systems MATHEMATICA and SINGULAR, [9], respectively. We see that in both cases the result is $\{y^3 - y^2, xy^2 - y^2, x^2 y - y^2, x^3 - y\}$.

```
In[6]:= GroebnerBasis[{-x^3 + y, x^2 y - y^2}, {x, y}]
Out[6]= {-y^2 + y^3, -y^2 + x y^2, x^2 y - y^2, x^3 - y}
```

Figure 3: *Output of Gröbner basis in system MATHEMATICA*

```
                     SINGULAR                                 /
 A Computer Algebra System for Polynomial Computations       /   version 3-1-
                                                           0<
 by: W. Decker, G.-M. Greuel, G. Pfister, H. Schoenemann     \   Jan 2012
FB Mathematik der Universitaet, D-67653 Kaiserslautern        \
> ring r1=0, (x,y),lp;
> poly f1=-x3+y;
> poly f2=x2*y-y2;
> ideal I=f1,f2;
> ideal G=groebner(I);
> G;
G[1]=y3-y2
G[2]=xy2-y2
G[3]=x2y-y2
G[4]=x3-y
>
```

Figure 4: *Output of Gröbner basis in system SINGULAR*

A reason to use more special systems than MATHEMATICA offers is to compute the Gröbner basis with respect to some special monomial order. In Figure 5 we compute the Gröbner basis of $\langle -x^3+y, x^2y-y^2\rangle$ using SINGULAR with respect to the weighted monomial order with weight vector $\vec{c}=(1,3)$. Note, that the result $G_{<_{(1,3)}}=\{x^3-y, y^2-x^2y\}$ is not the same as the Gröbner basis with respect to lexicohraphic monomial order $G_{<_{lex(y>x)}}=\{-x^5+x^6, -x^3+y\}$.

```
                     SINGULAR                                 /
 A Computer Algebra System for Polynomial Computations       /   version 3-1-4
                                                           0<
 by: W. Decker, G.-M. Greuel, G. Pfister, H. Schoenemann     \   Jan 2012
FB Mathematik der Universitaet, D-67653 Kaiserslautern        \

 > ring r1=0, (x,y),Wp(1,3);
 // ** redefining r1 **
 > poly f1=-x3+y;
 > poly f2=x2*y-y2;
 > ideal I=f1,f2;
 > ideal gI=groebner(I);
 > gI;
 gI[1]=x3-y
 gI[2]=y2-x2y
```

Figure 5: *Gröbner basis $G_{<_{(1,3)}}$ of $\langle x^3-y, -x^2y+y^2\rangle$ computed in SINGULAR*

Recall that to solve a system of linear equations, an effective method is to reduce it to the form in which an initial string of variables is missing from some of the equations, that is, the so-called "row-echelon" form. The next definition and theorem provide a way to eliminate a group of variables from a system of nonlinear polynomials. Moreover, it provides a way to find all solutions of a polynomial system in the case that the solution set is finite, or in other words, to find the variety of a polynomial ideal in the case that the variety is zero-dimensional.

For any ideal $I=\langle f_1,\dots,f_s\rangle \subset k[x_1,x_2,\dots,x_n]$ the l −th elimination ideal I_l is the ideal of $k[x_{l+1},x_{l+2},\dots,x_n]$ defined by

$$I_l = \mathrm{I} \cap k[x_{l+1}, x_{l+2}, \dots, x_n].$$

In the case of solving a system of nonlinear equations (1.1), this means that $I = \langle f_1, \dots, f_s \rangle$, but the elements of I_l are the equations (polynomials) that follow from $f_1 = 0, \dots, f_s = 0$ and eliminate the variables $x_1, \dots, x_l$. Concerning the Gröbner bases and elimination ideals, we have the following

Theorem, ([4]). If $G = \{g_1, \dots, g_t\}$ is a Gröbner basis for $I = \langle f_1, \dots, f_s \rangle \subset k[x_1, \dots, x_n]$ with respect to lexicographic order with $x_1 > \cdots > x_n$, then for each $0 \leq l \leq n$ the set

$$G_l = \mathrm{G} \cap k[x_{l+1}, x_{l+2}, \dots, x_n]$$

is a Gröbner basis for the l −th *elimination ideal*.

Gröbner basis theory allows one to find all solutions of a system (1.1) if the system has only a finite number of solutions. In such case a Gröbner basis with respect to the lexicographic order is always in a “row-echelon” form, as can be seen using the following example, [5]. Consider the polynomials

$$f_1 = x^2 + yz + x$$
$$f_2 = z^2 + xy + z$$
$$f_3 = y^2 + xz + y.$$

With respect to the lexicographic order with $x > y > z$, the Gröbner basis of ideal $\langle f_1, f_2, f_3 \rangle$ is $G = \{g_1, g_2, g_3, g_4, g_5, g_6\}$, where

$$g_1 = x + x^2 + yz$$
$$g_2 = xy + z + z^2$$
$$g_3 = z + xz + yz + z^2 + 2yz^2$$
$$g_4 = y + y^2 - z - yz - z^2 - 2yz^2$$
$$g_5 = z^2 + 2yz^2 + z^3 + 2yz^3$$
$$g_6 = z^2 + 3z^3 + 2z^4.$$

Thus, the system $f_1 = f_2 = f_3 = 0$ is equivalent to the system

$$g_1 = g_2 = g_3 = g_4 = g_5 = g_6 = 0.$$

For a generic system (1.1), a Gröbner basis may have significantly more complex structure than obtained in this example. However, if the system has only a finite number of solutions (i.e. the ideal is zero-dimensional), then any reduced Gröbner basis $\{g_1, \dots, g_n\}$ must contain a polynomial in one variable, let say, $g_1(x_1)$. Then, there is a subset of the Gröbner basis depending on this variable and one more variable, say, $g_2(x_1, x_2), \dots, g_t(x_1, x_2)$, etc. Thus, we first solve (perhaps only numerically) the equation $g_1(x_1) = 0$. Then, for each solution x_1^* of $g_1(x_1) = 0$, we find the solutions of $g_2(x_1^*, x_2) = \cdots = g_t(x_1^*, x_2) = 0$, which is a system of

polynomials in a single variable x_2. Continuing the process, we obtain in this way all solutions to (1.1). Thus, in the case of the finite number of solutions, Gröbner basis computations theoretically provide the complete solution to the problem (see e.g. Section 2.2. of [1] for more details).

3 GRÖBNER BASIS AND INTEGER LINEAR PROGRAMMING

In previous sections, we considered the process of multi-division and the Gröbner bases theory. Before the formal definition of the integer linear programming problem, note that the essence of the problem concerns the integer solutions of a linear system $A\vec{x} = \overrightarrow{b}$ (constraints), which optimizes the so-called cost function. Therefore, it would be quite convenient if the rows of the matrix equation $A\vec{x} = \vec{b}$ (i.e. the equations) could be presented (in the first approximation) as exponents of some new variables. In order to make things more straightforward, let us consider $A\vec{x} = \vec{b}$ (where: $[A]_{i,j} = a_{i,j}$ and $\vec{b} = (b_1, \dots, b_m)$) with the following restrictions: $a_{i,j} \in \mathbb{Z}$, $b_i \in \mathbb{Z}$ and $c_j \in \mathbb{R}$ with $i = 1,2, \dots, n$ and $j = 1,2, \dots, m$. We want to find a solution $\vec{x} = (x_1, x_2, \dots, x_n) \in \mathbb{Z}^n$ to

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n &= b_1 \\ &\vdots \\ a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n &= b_m, \end{aligned} \tag{3.1}$$

which minimizes the cost function $c(x_1, x_2, \dots, x_n) = \sum_{j=1}^{n} c_j x_j$. We call (3.1) an integer (linear) program (IP). In the matrix form, we have

$$\text{minimize } \vec{c} \cdot \overrightarrow{x}, \text{ subject to } A\vec{x} = \vec{b},$$

where $A \in \mathbb{Z}^{m\times n}$ *and* $\vec{b} = (b_1, \dots, b_m) \in \mathbb{Z}^m$. Note that all coefficients (including the solution vector) are generally allowed to be from the set $\mathbb{Z}$, but for now let the coefficients be limited to be natural numbers: $\mathbb{N}$.

The main mathematical idea which makes use of Gröbner bases when solving IP (3.1) is to associate new variables X_k for $k = 1,2, \dots, m$ (one variable to each equation) to (3.1) to represent the k −th equation in (3.1):

$$X_k^{a_{k1}x_1 + a_{k2}x_2 + \cdots + a_{kn}x_n} = X_k^{b_k} \leftarrow \mathrm{X_k} \text{ represents the } k - \text{th equaition}$$

The use of new variables makes the formulation of system (3.1) much simpler:

$$X_1^{a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n} \cdots X_m^{a_{m1}x_1 + a_{m2}x_2 + \cdots + a_{mn}x_n} = X_1^{b_1} \cdots X_m^{b_m},$$

which is equivalent to

$$\left(X_1^{a_{11}} \cdots X_m^{a_{m1}}\right)^{x_1} \cdot \ldots \cdot \left(X_1^{a_{1n}} \cdots X_m^{a_{mn}}\right)^{x_n} = \vec{X}^{\vec{b}}. \tag{3.1a}$$

Note that in (3.1a) the original (integer) variables X_k $(k = 1, \dots, n)$ adopt the meaning of the (integer) exponent of $X_1^{a_{1k}} \cdots X_m^{a_{mk}}$. Therefore, to each column of (3.1) or equivalently to each term in the brackets $(\dots)$ in (3.1a), we associate a new variable $Y_k = X_1^{a_{1k}} \cdots X_m^{a_{mk}}$ for each $k = 1,2, \dots, n$.
The *first step* in solving our IP problem is to determine whether a solution exists or not. The theory of Gröbner bases helps to characterize the existence and finally to prove the *optimality of the solutions* to IP (3.1).
The *crucial idea in solving* (3.1) in terms of Gröbner bases plays the following ring homomorphism

$$\Phi: k[Y_1, \dots, Y_n] \to k[X_1, \dots, X_m],$$

which associates to any polynomial from the ring $k[Y_1, \dots, Y_n]$ a polynomial from the ring $k[X_1, \dots, X_m]$, which is defined by:

$$\Phi(Y_k) = X_1^{a_{1k}} \cdots X_m^{a_{mk}}. \tag{3.2}$$

We immediately see that homomorphism (3.2) is in a one-to-one relation with a system of the form (3.1) (more precisely, the image of every Y_k is bijective related with the k-th column on the left side of (3.1)); furthermore, according to (3.1a), we have

$$\Phi\left(Y_1^{x_1} \cdot \dots \cdot Y_n^{x_n}\right) = \vec{X}^{\vec{b}}.$$

Concerning the map Φ defined by (3.2) we can assert the following. If we assume that all coefficient $a_{i,j}$ and b_i from (3.1) are non-negative, then a solution $\vec{x} = \tilde{x} \in \mathbb{N}_0^n$ of (3.1) exists if and only if the monomial $X_1^{b_1} \cdots X_m^{b_m}$ is in the image under homomorphism Φ. This means that a monomial $\vec{Y}^{\tilde{x}} \in k[Y_1, \dots, Y_n]$ exists for which $\Phi(\vec{Y}^{\tilde{x}}) = X_1^{b_1} \cdots X_m^{b_m}$, and $\tilde{x} \in \mathbb{N}_0^n$ is a solution to (3.1).

Both implications of the above assertion can be proved at once. Let $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n) \in \mathbb{N}_0^n$ be a solution of (3.1). This means that (3.1a) holds for $\vec{b} = \tilde{x}$. According to definition of Φ this is equivalent to

$$(\Phi(Y_1))^{\tilde{x}_1} \cdots (\Phi(Y_n))^{\tilde{x}_n} = X_1^{b_1} \cdots X_m^{b_m}$$

Now, since Φ is a homomorphism we have

$$\Phi(Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}) = X_1^{b_1} \cdots X_m^{b_m}$$

and $X_1^{b_1} \cdots X_m^{b_m}$ is in the image under Φ and the corresponding original is $Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n} \in k[Y_1, \ldots, Y_n]$, and we have shown that this is equivalent to $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n) \in \mathbb{N}_0^n$ being solution to (3.1).

This means that $\tilde{x}$ for which $\vec{Y}^{\tilde{x}}$ is in the image under Φ is an integer solution to system (3.1), but still we have to handle the problem of the minimality condition of the cost function $c(x_1, x_2, \ldots, x_n) = \sum_{j=1}^n c_j x_j$. In order to solve this, we need some additional results. First, if $x_1, x_2, \ldots, x_n$ and $y_1, y_2, \ldots, y_n$ are elements of a commutative ring k, then for any non-negative integers $\alpha_1, \alpha_2, \ldots, \alpha_n$ the polynomial $x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n} - y_1^{\alpha_1} y_2^{\alpha_2} \cdots y_n^{\alpha_n}$ is in the ideal $\langle x_1 - y_1, \ldots, x_n - y_n \rangle$. The proof can be done by mathematical induction on the number of elements x_n and y_n. Let us simply mention that for $n = 1$ (the basis for induction) the assertion is a well-known result

$$x^\alpha - y^\alpha = (x - y)(x^{\alpha-1} + x^{\alpha-2}y + \cdots + xy^{\alpha-2} + y^{\alpha-1}).$$

For the rest of the proof, see [12].
Now let $K = \langle Y_1 - f_1, \ldots, Y_n - f_n \rangle \subseteq k[Y_1, \ldots, Y_n, X_1, \ldots, X_m]$ and $f_n \in k[X_1, \ldots, X_m]$. If $g \in K \cap k[Y_1, \ldots, Y_n]$, then $g(Y_1, \ldots, Y_n) = \sum_{i=1}^n (Y_i - f_n) h_i$, where $h_i \in k[Y_1, \ldots, Y_n, X_1, \ldots, X_m]$. Let Φ map $Y_i \longmapsto f_i$. Then $g(f_1, \ldots, f_n)$=0, thus $\Phi(g) = 0$. In contrast, let $\Phi(g) = 0$ and let $g = \sum_\alpha c_\alpha Y_1^{\alpha_1} Y_2^{\alpha_2} \cdots Y_n^{\alpha_n}$ (where finitely many $c_\alpha \neq 0$). Since $g(f_1, \ldots, f_n)$=0, we have

$$g = g - 0 = g - g(f_1, \ldots, f_n)$$

$$g = \sum_\alpha c_\alpha Y_1^{\alpha_1} Y_2^{\alpha_2} \cdots Y_n^{\alpha_n} - \sum_\alpha c_\alpha f_1^{\alpha_1} f_2^{\alpha_2} \cdots f_n^{\alpha_n}$$
$$g = \sum_\alpha c_\alpha \left(Y_1^{\alpha_1} Y_2^{\alpha_2} \cdots Y_n^{\alpha_n} - f_1^{\alpha_1} f_2^{\alpha_2} \cdots f_n^{\alpha_n} \right)$$

which is in the ideal $K = \langle Y_1 - f_1, \ldots, Y_n - f_n \rangle$, according to the previous result. Thus, $Ker(\Phi) = K \cap k[Y_1, \ldots, Y_n]$, and the elements of $Ker(\Phi)$ can (by the elimination theorem) be found in the following way: first find the Gröbner basis of the ideal $K = \langle Y_1 - f_1, \ldots, Y_n - f_n \rangle$ in $k[Y_1, \ldots, Y_n, X_1, \ldots, X_m]$ with respect to a lex ordering $X_1 > \cdots > X_m > Y_1 > \cdots > Y_n$. Then a Gröbner basis for $K \cap k[Y_1, \ldots, Y_n]$ will be precisely the polynomials of the Gröbner basis of K that do not have any X variables. Obviously, for a given homomorphism Φ, any monomial $f \in k[Y_1, \ldots, Y_n]$ is not in the image of Φ. Let us denote the Gröbner basis of $K = \langle Y_1 - f_1, \ldots, Y_n - f_n \rangle$ obtained by the elimination theorem by $\tilde{G}$. According to the above results, we have

$$f \in Ker(\Phi) \Leftrightarrow \exists h \in k[Y_1, \ldots, Y_n] \quad s.t. \quad f \xrightarrow{\tilde{G}} h,$$

which is then the key to the solution of IP (3.1) constrained by the cost function $c(x_1, x_2, \ldots, x_n) = \sum_{j=1}^n c_j x_j$. Finally note that, if take care that the monomial order which is used to compute the Gröbner basis, $\tilde{G}_{<_c}$, of the ideal $K = \langle Y_1 - f_1, \ldots, Y_n - f_n \rangle$ is compatibile with the cost function $c(x_1, x_2, \ldots, x_n) = \sum_{j=1}^n c_j x_j$ and with the corresponding

$$\Phi\left(Y_1^{x_1} \cdots Y_n^{x_n}\right) = \Phi\left(Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}\right)$$

and

$$c(x_1, x_2, \ldots, x_n) < c(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n)$$

$$\Downarrow$$

$$Y_1^{x_1} \cdots Y_n^{x_n} <_c Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}$$

then from $X_1^{b_1} \cdots X_m^{b_m} \xrightarrow{\tilde{G}_{<_c}} Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}$ one can deduce that $\tilde{x} = (\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n) \in \mathbb{N}_0^n$ is a solution to IP (3.1) for which the cost function $c(\vec{x}) = \vec{c} \cdot \vec{x}$ *is minimal.* The minimality of the solution is proven by contradiction. If any other solution $\vec{x} = (x_1, x_2, \ldots, x_n)$ is minimal then, since $c(\vec{x}) = c(\tilde{x})$ and since the term ordering $<_c$ is compatibile with the cost function and with the corresponding system we have $\Phi\left(Y_1^{x_1} \cdots Y_n^{x_n}\right) = \Phi\left(Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}\right)$; implying $\Phi\left(Y_1^{x_1} \cdots Y_n^{x_n} - Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n}\right) = 0$, which means $Y_1^{x_1} \cdot\cdot Y_n^{x_n} - Y_1^{\tilde{x}_1} \cdots Y_n^{\tilde{x}_n} \xrightarrow{\tilde{G}_{<_c}} 0$, which contradicts the assumption that $Y_1^{x_1} \cdots Y_n^{x_n}$ is already reduced with respect to $\tilde{G}_{<_c}$.

The above results are the basis for Conti & Traverso's well-known algorithm, which is described below (see ([3]) for more details). However, first we have to consider how to transform (3.1) which can contain some negative integers; recall that generally $a_{i,j} \in \mathbb{Z}$ and $b_i \in \mathbb{Z}$. This can be generally transformed to an IP with strictly nonnegative (integer) coefficients $a_{i,j}, b_i$ by adding an extra indeterminate W defined by

$$X_1 \cdot X_2 \cdot \cdots \cdot X_m \cdot W = 1, \tag{3.3}$$

which transforms

$$\vec{X}^{Aj} := X_1^{a_{1j}} \cdot \ldots \cdot X_i^{-a_{ij}} \cdot \ldots \cdot X_m^{a_{mj}}$$

to

$$X_1^{a_{1j}+a_{ij}} \cdot \ldots \cdot X_i^{0} \cdot \ldots \cdot X_m^{a_{mj}+a_{ij}} \cdot W^{a_{ij}} = \vec{X}^{Aj} W_j,$$

where $W_j = W^{a_{ij}}$. In a similar way, if there are some negative entries in $\vec{b}$, we transform $\vec{X}^{\vec{b}}$ to $\vec{X}^{\vec{b}} W_{\vec{b}}$.

The optimal solution of IP (3.1) with some negative integers is, therefore, obtained in the following way:

- Define W by (3.3), if there are some negative entries in $A, \vec{b}$
- define an ideal $I = \left\{Y_1 - \vec{X}^{A_1}, \ldots, Y_n - \vec{X}^{A_n}\right\}$ on the polynomial ring $k[X_1, \ldots, X_m, Y_1, \ldots, Y_n]$, if there are no negative entries in $A, \vec{b}$

- define an ideal $I = \{Y_1 - \dot{X}^{A_1}W_1, \dots, Y_n - \dot{X}^{A_n}W_n, X_1 \cdot X_2 \cdot \dots \cdot X_m \cdot W - 1\}$ on the polynomial ring $k[X_1, \dots, X_m, W, Y_1, \dots, Y_n]$, if there are some negative entries in $A, \vec{b}$
- let G be a reduced Gröbner basisof $I$with respect to a monomial order $<_{\vec{c}}$, where $\vec{c}$ is defined by the cost function $\vec{c} \cdot \vec{x}$
- dividing $\vec{X}^{\vec{b}}W_{\vec{b}}$ (i.e. the generalization of $\vec{X}^{\vec{b}}$) by G always yields a remainder $R \in k[Y_1, \dots, Y_n]$, which because of its minimality (ensured by the multivariable division algorithm) ensures the optimality of the solution; thus the solution $\vec{x} = (\beta_1, \dots, \beta_n)$ to IP (3.1) is obtained by reducing $\vec{X}^{\vec{b}}W_{\vec{b}}$ by G which yields a remainder $R = {Y_1}^{\beta_1} \cdots {Y_n}^{\beta_n}$ and thereby the solution $\vec{x} = (\beta_1, \dots, \beta_n)$.

The next example will show the method for determining whether system of the form (3.1) has a non-negative integer solution, and for finding a solution. The method consists of the following three steps, [12]:

- Compute a Gröbner basis G for the ideal $K = \langle Y_j - X_1^{a_{1j}} \cdots X_m^{a_{mj}} : 1 \le j \le n \rangle$ with respect to an elimination order with the X variables greater than Y variables.
- Find the remainder r of the division of the monomial $X_1^{b_1} \cdots X_m^{b_m}$ by G.
- If $r \notin k[Y_1, \dots, Y_n]$, then system (3.1) does not have non-negative integer solutions. If $r = Y_1^{x_1} \cdots Y_n^{x_n}$, then $(x_1, \dots, x_n)$ is a solution of system (3.1), [1].

Now, we show how the proposed method works.

Example 1: Let us check if there exist non-integer solutions of system

$$2x_1 + x_2 = 3$$
$$x_1 + x_2 + 3x_3 = 5.$$

On the first step, we compute a Gröbner basis G of an ideal $K = \langle Y_1 - X_1^2X_2, Y_2 - X_{1X_2}, Y_3 - X_2^3 \rangle$ with respect to lexicographic order with $X_1 > X_2 > Y_1 > Y_2 > Y_3$. We obtain

$$G = \{-{Y_2}^6 + {Y_1}^3Y_3, X_2{Y_2}^4 - {Y_1}^2Y_3, X_2Y_1 - {Y_2}^2, {X_2}^2{Y_2}^2 - Y_1Y_3, {X_2}^3 - Y_3, -{X_2}^2Y_2 + X_1Y_3, -Y_1 + X_1Y_2, X_1X_2 - Y_2\}.$$

Then, we divide monomial $X_1^3X_2^5$ by G and obtain $Y_1^1Y_2^1Y_3^1$. Therefore, the non-negative integer solution is $(x_1, x_2, x_3) = (1,1,1)$.

Example 2 ([8]): We show how to optimize the cost function with respect to some constraints $A\vec{x} = \vec{b}$, and the coefficients are now allowed to be also negative integers. Following (3.1), we have to minimize the cost function

$$\vec{c} \cdot \vec{x} = 1000x_1 + x_2 + x_3 + 100x_4$$

subject to

$$3x_1 - 2x_2 + x_3 - x_4 = -1$$
$$4x_1 + x_2 - x_3 = 5.$$

The solution to the above example obtained with system SINGULAR is shown in Figure 6. Note that the weighted term order is used with $\vec{C} = (1000002,1000001,1000000,1000,1,1,100)$ to ensure that $X1 > X2 > W > Y1 > Y2 > Y3 > Y4$ and to ensure the weight order $(1000,1,1,100)$, corresponding to $\vec{c} = (1000,1,1,100)$. Note that, for example, the monomials $\vec{X}^{\vec{b}}W_{\vec{b}}$ and $\vec{X}^{A_2}W_2$ are:

$$\vec{X}^{\vec{b}}W_{\vec{b}} = X_1^{-1}X_2^5 = X_1^{-1}X_2^{-1} \cdot X_2^1X_2^5 = W^1X_2^6,$$
$$\vec{X}^{A_2}W_2 = X_1^{-2}X_2^1 = X_1^{-2}X_2^{-2} \cdot X_2^2X_2^1 = W^2X_2^3.$$

The optimal solution $\vec{x} = (1,3,2,0)$ is obtained from the result of the multivariable division:

$$W^1\ X_2^6 \xrightarrow{G} Y_1^1Y_2^3Y_3^2Y_4^0.$$

```
                     SINGULAR                                 /
 A Computer Algebra System for Polynomial Computations       /   version 3-
                                                            0<
 by: W. Decker, G.-M. Greuel, G. Pfister, H. Schoenemann     \   Jan 2012
FB Mathematik der Universitaet, D-67653 Kaiserslautern        \
> ring r1=0,(X1,X2,W,Y1,Y2,Y3,Y4),Wp(1000002,1000001,1000000,1000,1,1,100);
> poly f1=Y1-X13*X24;
> poly f1=Y1-X1^3*X2^4;
> poly f2=Y2-X2^3*W^2;
> poly f3=Y3-X1^2*W;
> poly f4=Y4-X2*W;
> poly f5=X1*X2*W-1;
> ideal I=f1,f2,f3,f4,f5;
> ideal gI=groebner(I);
> reduce(W*X2^6,gI);
Y1*Y2^3*Y3^2
>
```

Figure 6: *Computing the optimal solution in system SINGULAR*

References

[1] **W.W. Adams, P. Loustaunau:** An introduction to Gröbner bases: Graduate Studies in Mathematics. Vol. 3, Providence, RI: American Mathematical Society, 1994

[2] **B. Buchberger:** Ein Algorithmus zum Auffinden der Basiselemente des Restlasseringes nach einem nulldimensionalen Polynomideal. PhD Thesis, Mathematical Institute, University of Innsbruck, Austria, 1965

[3] **P. Conti, C. Traverso:** Buchberger algorithm and integer programming, Applied algebra, falgebraic algorithms and error-correcting codes (New Orleans, LA, 1991), Lecture Notes in Comput. Sci. vol. 539, p. 130-139, 1991

[4] **D. Cox, J. Little, D. O'Shea: I**deals, Varieties and Algorithms: An Introduction to Computational Algebraic Geomety and Commutative Algebra. New York: Springer, 2007

[5] **S.R. Czapor:** Gröbner basis methods for solving algebraic equations. Ph.D Thesis. University of Waterloo, Canada, 1988

[6] **V.F. Edneral, A. Mahdi, V.G. Romanovski, D.S. Shafer:** The center problem on a center manifold in R3, Nonlinear Anal., vol. 75, p. 2614-2622, 2012

[7] B. Ferčec, M. Mencinger: Isochronicity of centers at a center manifold, AIP conference proceedings, 1468. Melville, N.Y.: American Institute of Physics, p. 148-157, 2012

[8] **S. Flory, E. Michel:** Integer Programming with Gröbner basis. (http://www.iwr.uni-heidelberg.de/groups/amj/People/Eberhard.Michel/Documents/Else/DiscreteOptimization.pdf)

[9] **G.M. Greuel, G. Pfister, H. Schönemann:** Singular 3.0. A Computer Algebra System for Polynomial Computations, Centre for Computer Algebra, University of Kaiserslautern, 2005; http://www.singular.uni-kl.de.

[10] **V.G. Romanovski, M. Mencinger, B. Ferčec:** Investigation of center manifolds of 3-dim systems using computer algebra. Program. comput. softw., vol. 39, no. 2, p. 67-73, 2013

[11] **V.G. Romanovski, D.S. Shafer:** The center and cyclicity problems: A computational algebra approach. Boston: Birkhauser Verlag, 2009

[12] **C. Wendler:** Groebner Bases with an Application to Integer Programming; (http://documents.kenyon.edu/math/CWendler.pdf), 2004