2006

Presentations of Finitely Generated Cancellative Commutative Monoids and Nonnegative Solutions of Systems of Linear Equations

Scott T. Chapman
Trinity University, schapman@trinity.edu

Pedro A. García Sánchez

David Llena

José Carlos Rosales

Available online at www.sciencedirect.com
SCIENCE DIRECT
Discrete Applied Mathematics 154 (2006) 1947–1959
DISCRETE APPLIED MATHEMATICS
www.elsevier.com/locate/dam

Presentations of finitely generated cancellative commutative monoids and nonnegative solutions of systems of linear equations

S.T. Chapman[a,1], P.A. García-Sánchez[b,2], D. Llena[c], J.C. Rosales[d,2]

[a] *Department of Mathematics, Trinity University, One Trinity Place, San Antonio, TX 78212-7200, USA*
[b] *Departamento de Álgebra, Universidad de Granada, 18071 Granada, Spain*
[c] *Departamento de Geometría, Topología y Química Orgánica, Universidad de Almería, La Cañada de San Urbano s/n, 04120 Almería, Spain*
[d] *Departamento de Álgebra, Universidad de Granada, 18071 Granada, Spain*

Received 16 September 2005; received in revised form 27 March 2006; accepted 27 March 2006
Available online 9 May 2006

Abstract

Varying methods exist for computing a presentation of a finitely generated commutative cancellative monoid. We use an algorithm of Contejean and Devie [An efficient incremental algorithm for solving systems of linear diophantine equations, Inform. and Comput. 113 (1994) 143–172] to show how these presentations can be obtained from the nonnegative integer solutions to a linear system of equations. We later introduce an alternate algorithm to show how such a presentation can be efficiently computed from an integer basis.

MSC: 20M14; 20M15

Keywords: Finitely generated commutative cancellative monoid; Presentation; System of linear equations

0. Introduction

Several approaches can be found in the mathematical literature to the problem of computing a presentation of a given finitely generated cancellative commutative monoid. These different approaches can be mainly divided into three different groups.

The implementations of the first group are based on the computation of the kernel of a ring morphism. This computation can be performed, after a trick that enables us to work with torsion free monoids, using the implicitation algorithm or using specializations of this algorithm to the case of morphisms between the ring of polynomials and the semigroup ring of a finitely generated commutative cancellative and torsion free monoid (see [6, 22]; in [9] a realization based on the algorithm appearing in [6] is given for Maple). The efficiency of these types of implementations depends on the efficiency of the computation of a Gröbner basis. The advantage of using this group of implementations is that they are easy to implement in any of the existing programmable software packages that include Gröbner basis computations

[1] The first author completed part of this work while on an academic leave granted by the Trinity University Faculty Development Committee.
[2] The second and fourth authors are supported by the project MTM2004-01446 and FEDER funds.
E-mail addresses: schapman@trinity.edu (S.T. Chapman), pedro@ugr.es (P.A. García-Sánchez), dllena@ual.es (D. Llena), jrosales@ugr.es (J.C. Rosales).

0166-218X/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.dam.2006.03.013

(singular, Maple V, Mathematica, etc.). The underlying idea is to eliminate several auxiliary variables used to define the kernel homomorphism. By using the same elimination procedure, in [18] an algorithm to compute the presentation of any finitely generated commutative submonoid of a finitely generated monoid is given.

The algorithms of the second group are based on a generalization of the algorithm appearing in [15] for the computation of a minimal presentation of a numerical semigroup. Though the enactment of this algorithm is highly efficient, neither the generalization appearing in [1] nor the generalization introduced in [19] have yielded efficient implementations of the problem (our students have implemented the algorithm described in [19], but the resulting software is much slower than the one appearing in [9]).

The last approach was proposed in [17]. The cancellative law implies linearization as we will see in Section 1, and thus the existing algorithms for finding the set of nonnegative integer solutions to linear systems of equations can be used. Again, several approaches exist for this problem: some use Dickson's lemma and Gröbner bases (see for instance [10]), others Elliot's trick (see [7]) or a generalization of Clausen–Fortenbacher's ([3]) geometrical point of view (see [4]). As a consequence of the nature of the problem, the number of variables needed to compute a presentation becomes considerably large. Roughly speaking, this is due to the fact that in order to describe a single integer you need two nonnegative integers.

Our work in this paper is organized into three sections. In Section 2 we use the algorithm presented in [4] to illustrate how nonnegative integer solutions to linear systems of equations yield presentations for finitely generated monoids. The systems of equations appearing are quite special, and the algorithms existing in the literature do not take advantage of this (obviously this is because they were studied for other purposes). In Section 3 we introduce an alternative method to compute presentations starting from an integer basis and thus no "duplication" of the number of variables is needed. The algorithm is easy to implement and it is based on the critical pair completion idea. In Section 1, we review for the reader the definitions, notation and basic results which are used in Sections 2 and 3.

1. Systems of equations and presentations of cancellative monoids

In this section, we recall some known facts about finitely generated cancellative monoids. These results can be found in [17, Chapter 8].

A *congruence* σ on $\mathbb{N}^n$ is an equivalence relation which is compatible with addition, that is to say, if (a, b) and (c, d) are elements in σ, then $(a + c, b + d)$ is in σ. Hence, σ is a submonoid of $\mathbb{N}^n \times \mathbb{N}^n$, since we also have that $(0, 0) \in \sigma$.

A congruence does not necessarily need to be finitely generated as a monoid. This contrasts with the fact that every congruence, as a congruence, is finitely generated (see [13]).

Example 1. Let σ be the congruence on $\mathbb{N} = \{0, 1, 2, \ldots\}$ defined by

$$x\,\sigma\,y \quad \text{if} \begin{cases} x \geqslant 1 \text{ and } y \geqslant 1, \\ \text{or} \\ x = y \text{ otherwise.} \end{cases}$$

It is easy to show that $\{(x, 1) \mid x \geqslant 1\}$ is contained in σ and that the elements of this set cannot be expressed as a sum of two other nontrivial elements of σ. Therefore σ cannot be finitely generated as a monoid.

There are many congruences on $\mathbb{N}^n$ that are finitely generated as monoids. As we see next, every cancellative congruence satisfies this condition. A congruence on $\mathbb{N}^n$ is *cancellative* if and only if $(a + c)\sigma(b + c)$ implies $a\,\sigma\,b$ (this is equivalent to the fact that $\mathbb{N}^n/\sigma$ is a cancellative monoid). If σ is a cancellative congruence and $a\,\sigma\,b$, then we can eliminate the "common" part of a and b and substitute by a' and b' so that $a = a' + c$, $b = b' + c$, for some $c \in \mathbb{N}^n$ and $a' \cdot b' = 0$ (where $(x_1, \ldots, x_n) \cdot (y_1, \ldots, y_n) = x_1y_1 + \cdots + x_ny_n$). In this way, we can "codify" the information contained in the assertion $a\,\sigma\,b$ by $a' - b' = a - b$. This idea motivates the following definitions. The Abelian group associated to a congruence σ on $\mathbb{N}^n$ is the subgroup of $\mathbb{Z}^n$ defined by

$$M_\sigma = \{a - b \in \mathbb{Z}^n \mid (a, b) \in \sigma\}.$$

Conversely, for a given subgroup H of $\mathbb{Z}^n$, we define the following congruence of $\mathbb{N}^n$

$$\sim_H = \{(a, b) \in \mathbb{N}^n \mid a - b \in H\}.$$

It is easy to show that $\sim_H$ is always cancellative and that a congruence σ on $\mathbb{N}^n$ is cancellative if and only if $\sigma = \sim_{M_\sigma}$. Therefore, studying cancellative congruences is equivalent to studying congruences of the form $\sim_H$, with H a subgroup of $\mathbb{Z}^n$.

Proposition 2 (*Rosales and García-Sánchez [17, Proposition 8.1]*). *For every subgroup M of $\mathbb{Z}^n$, the congruence $\sim_M$ is a finitely generated submonoid of $\mathbb{N}^n \times \mathbb{N}^n$.*

In the proof of this result, it is shown that $\sim_M$ is generated by the minimal elements of $\sim_M$ with respect to the usual partial order, which are called the *irreducibles* of $\sim_M$. The set of these elements will be denoted by $\mathfrak{I}(\sim_M)$, and it is finite due to Dickson's Lemma (in $\mathbb{N}^n$ the usual partial order is defined by $(x_1, \ldots, x_n) \leqslant (y_1, \ldots, y_n)$ if $x_i \leqslant y_i$ for all i).

If the set

$$\rho = \{(a_1, b_1), \ldots, (a_t, b_t)\}$$

is a system of generators of σ as a monoid, then it is also a system of generators of σ as a congruence. This is due to the fact that for $(a, b) \in \sigma$, there exists $i_1, \ldots, i_s \in \{1, \ldots, t\}$ such that

$$(a, b) = (a_{i_1}, b_{i_1}) + \cdots + (a_{i_s}, b_{i_s}),$$

and since the congruence generated by ρ is also a monoid, we obtain that (a, b) is in the congruence generated by ρ. Observe that we can eliminate from ρ the elements of the form (a, a), since these elements are always in the congruence generated by ρ (they are always in the reflexive closure). Analogously, if (a, b) and (b, a) are both in ρ, then we can eliminate one of them. This motivates the definition of primitive element. A *primitive* element (a, b) of a congruence $\sim_M$ is an element of $\mathfrak{I}(\sim_M)$ such that $a \neq b$. By denoting by $\mathfrak{P}(\sim_M)$ the set of primitive elements of $\sim_M$, we have the following consequence.

Corollary 3 (*Rosales and García-Sánchez [17, Proposition 8.3]*). *Let M be a subgroup of $\mathbb{Z}^n$. The congruence $\sim_M$ is generated, as a congruence, by $\mathfrak{P}(\sim_M)$.*

We introduce the following notation for the rest of the paper. Let $p, r, k \in \mathbb{N}$, let $d_1, \ldots, d_r$ be positive integers and let $m_i = (a_{1i}, \ldots, a_{(r+k)i})$, $i \in \{1, \ldots, p\}$. Let S be the submonoid of $\mathbb{Z}_{d_1} \times \cdots \times \mathbb{Z}_{d_r} \times \mathbb{Z}^k$ that is generated by $\{m_1, \ldots, m_p\}$.

Then we have a monoid morphism

$$\varphi : \mathbb{N}^p \to S, \quad \varphi(x_1, \ldots, x_p) = \sum_{i=1}^{p} x_i m_i.$$

It is easy to see [14] that the kernel of this morphism is of the form $\sim_M$, where M is the subgroup of $\mathbb{Z}^p$ whose elements $(x_1, \ldots, x_p)$ satisfy the equations

$$\begin{aligned}
a_{11}x_1 + \cdots + a_{1p}x_p &\equiv 0 \,(\mathrm{mod}\, d_1),\\
&\vdots\\
a_{r1}x_1 + \cdots + a_{rp}x_p &\equiv 0 \,(\mathrm{mod}\, d_r),\\
a_{(r+1)1}x_1 + \cdots + a_{(r+1)p}x_p &= 0,\\
&\vdots\\
a_{(r+k)1}x_1 + \cdots + a_{(r+k)p}x_p &= 0.
\end{aligned}$$

Hence $S \cong \mathbb{N}^p/\sim_M$. Every commutative finitely generated cancellative monoid is isomorphic to some monoid S as above (see for instance [17, 20]).

A *presentation* of S is a system of generators (as a congruence) of $\sim_M$. Hence, by Corollary 3, for computing a presentation of S, it suffices to compute $\mathfrak{P}(\sim_M)$. Calculating the set of primitive elements of $\sim_M$ is equivalent to finding

the elements $x \in M\backslash\{0\}$ such that there exists no $y \in M\backslash\{0, x\}$ satisfying that $(y^+, y^-) \leqslant (x^+, x^-)$, where

$$(a_1, \ldots, a_p)^+ = (\max\{a_1, 0\}, \ldots, \max\{a_p, 0\}),$$

$$(a_1, \ldots, a_p)^- = (-\min\{a_1, 0\}, \ldots, -\min\{a_p, 0\}).$$

Note that $x = (x_1, \ldots, x_p) \in M$ is one of these elements if and only if $x^+ + x^-$ is a minimal nonnegative nontrivial integer solution of the system of equations

$$\begin{aligned}
\varepsilon_1 a_{11} x_1 + \cdots + \varepsilon_p a_{1p} x_p p &\equiv 0 \,(\mathrm{mod}\, d_1),\\
&\vdots\\
\varepsilon_1 a_{r1} x_1 + \cdots + \varepsilon_p a_{rp} x_p &\equiv 0 \,(\mathrm{mod}\, d_r),\\
\varepsilon_1 a_{(r+1)1} x_1 + \cdots + \varepsilon_p a_{(r+1)p} x_p &= 0,\\
&\vdots\\
\varepsilon_1 a_{(r+k)1} x_1 + \cdots + \varepsilon_p a_{(r+k)p} x_p &= 0,
\end{aligned}$$

where

$$\varepsilon_i = \begin{cases} 1 & \text{if } x_i \geqslant 0,\\ -1 & \text{otherwise.} \end{cases}$$

Hence there exists a tight connection between primitive elements of cancellative congruences and minimal nonnegative nontrivial elements of a certain subgroup of $\mathbb{Z}^p$.

The degree (sum of all its coordinates) of a primitive element is always bounded, and a bound can be obtained as follows. From Pottier's bound [11] one easily obtains [17, Theorem 7.5], which is the analog of Pottier's bound for systems in which some of the equations can be in congruences. The idea is to convert an equation of the form $a_1x_1 + \cdots + a_nx_n \equiv 0 \bmod b$ to $a_1x_1 + \cdots + a_nx_n + \varepsilon by = 0$, and then apply Pottier's bound. By using this together with the above remark, we obtain the following consequence.

Proposition 4 (*Rosales and García-Sánchez [17, Corollary 8.8]*). *Let $a = (a_1, \ldots, a_p)$ and $b = (b_1, \ldots, b_p)$ be elements of $\mathbb{N}^p$. If (a, b) is a primitive element of $\sim_M$, then*

$$a_1 + \cdots + a_p + b_1 + \cdots + b_p \leqslant \prod_{i=1}^{r} \left(1 + |d_i| + \sum_{j=1}^{p} |a_{ij}|\right) \prod_{i=r+1}^{r+k} \left(1 + \sum_{j=1}^{p} |a_{ij}|\right).$$

Since the set defined by this inequality is finite, we can compute the set of minimal elements (with respect to $\leqslant$) of $\sim_M\backslash\{(0, 0)\}$ and consequently we can compute a system of generators of $\sim_M$. This result is more important from a theoretical point of view than from a practical one. An exhaustive search in the region defined by the inequality of Proposition 4 is very inefficient.

The bound given in Proposition 4 can be used in a different manner, which is related to the first group of implementations mentioned in the introduction. Let y be a symbol. Define $K[S] = \bigoplus_{s\in S} K y^s$. Addition in $K[S]$ is performed componentwise, and multiplication is accomplished following the rule $y^s y^t = y^{s+t}$ (together with de distributive law). The set $K[S]$ becomes a commutative ring in this way, and the monoid morphism $\varphi : \mathbb{N}^p \to S$ induces a ring morphism

$$\varphi : K[x_1, \ldots, x_p] \to K[S], \quad \varphi(x_i) = y^{m_i}.$$

Let $I_{\sim_M}$ be the kernel of this morphism. Herzog shows in [8] that

$$\{(a_1, b_1), \ldots, (a_t, b_t)\}$$

is a system of generators of $\sim_M$ (as a congruence) if and only if $I_{\sim_M}$ is generated by

$$\{X^{a_1} - X^{b_1}, \ldots, X^{a_t} - X^{b_t}\},$$

where $X^{(n_1,\ldots,n_p)} = x_1^{n_1}\cdots x_p^{n_p}$. Moreover,

$$\begin{aligned} I_{\sim_M} &= \langle X^a - X^b \in K[x_1,\ldots,x_p] \,|\, (a,b) \in \sim_M \rangle \\ &= \langle X^a - X^b \in K[x_1,\ldots,x_p] \,|\, a - b \in M \rangle \end{aligned}$$

and $(a,b) \in \sim_M$ if and only if $X^a - X^b \in I_{\sim_M}$.

If $X^a - X^b$ belongs to a (reduced) Gröbner basis of $I_{\sim_M}$, then there exists no $(c,d) \in \sim_M$ such that $(c,d) < (a,b)$. This implies that (a,b) is a primitive element of $\sim_M$. Therefore we obtain the following consequence.

Corollary 5. *Let $a = (a_1,\ldots,a_p)$ and $b = (b_1,\ldots,b_p)$ be elements of $\mathbb{N}^p$. If $X^a - X^b$ belongs to a reduced Gröbner basis of $I_{\sim_M}$, then*

$$a_1 + \cdots + a_p + b_1 + \cdots + b_p \leqslant \prod_{i=1}^{r}\left(1 + |d_i| + \sum_{j=1}^{p}|a_{ij}|\right)\prod_{i=r+1}^{r+k}\left(1 + \sum_{j=1}^{p}|a_{ij}|\right).$$

This provides a bound for the (total) degree of the elements belonging to a Gröbner basis of the ideal $I_{\sim_M}$. This bound is different from (and not comparable to) the bound introduced in [21,22], for the case where S is also torsion free (and therefore $I_{\sim_M}$ is a prime ideal). A nice comparison of different bounds and algorithms for solving linear Diophantine equations over the set of non-negative integers can be found in [12].

2. Computing a minimal presentation of a finitely generated cancellative monoid

Let S and M be as in the preceding section. As we have mentioned before, in order to find a presentation of S, it suffices to compute the set $\mathfrak{P}(\sim_M)$. Clearly, if $((a_1,\ldots,a_p),(b_1,\ldots,b_p))$ is a primitive element of $\sim_M$, then $(a_1,\ldots,a_p,b_1,\ldots,b_p)$ is a minimal nonnegative nontrivial integer solution of the system of equations:

$$\begin{aligned} a_{11}x_1 + \cdots + a_{1p}x_p - a_{11}x_{p+1} - \cdots - a_{1p}x_{2p} &\equiv 0 \,(\mathrm{mod}\, d_1), \\ &\vdots \\ a_{r1}x_1 + \cdots + a_{rp}x_p - a_{r1}x_{p+1} - \cdots - a_{rp}x_{2p} &\equiv 0 \,(\mathrm{mod}\, d_r), \\ a_{(r+1)1}x_1 + \cdots + a_{(r+1)p}x_p - a_{(r+1)1}x_{p+1} - \cdots - a_{(r+1)p}x_{2p} &= 0, \\ &\vdots \\ a_{(r+k)1}x_1 + \cdots + a_{(r+k)p}x_p - a_{(r+k)1}x_{p+1} - \cdots - a_{(r+k)p}x_{2p} &= 0. \end{aligned}$$

By using the results appearing in [16], we can transform the congruences into linear homogeneous equations. Then we can use the algorithm appearing in [4] in order to find the minimal nonnegative nontrivial integer solutions of this system of equations. This transformation is analogous to the one already given above to obtain bounds for the primitive elements. The idea consists of replacing $a_1x_1 + \cdots + a_nx_n \equiv 0 \mod b$ with $a_1x_1 + \cdots + a_nx_n + by_1 - by_2 = 0$. This yields two new variables for each congruence in the original system. Once we obtain the minimal solutions to the new system, we project onto the original variables. The resulting set is $\mathfrak{I}(\sim_M)$.

Note that we have used a different system of the equations for obtaining the bound of Proposition 4. This is due to the fact that if we use this latter system of equations, then the bound is worse than the one obtained in Proposition 4, since the number of unknowns is twice the number of unknowns appearing in the system used to obtain the mentioned bound.

Let us illustrate the procedure for computing $\mathfrak{P}(\sim_M)$ with a few examples.

Example 6. Let S be the submonoid of $\mathbb{N}^2$ generated by

$$\{(1,2),(0,3),(1,1),(2,3)\}.$$

Clearly S is a cancellative commutative monoid, since it is a submonoid of $\mathbb{N}^2$. The Abelian group M is the subgroup of $\mathbb{Z}^4$ whose defining (the columns of these equations are the generators of S) are

$$x_1 + x_3 + 2x_4 = 0,$$

$$2x_1 + 3x_2 + x_3 + 3x_4 = 0.$$

The monoid S is isomorphic to $\mathbb{N}^4/\sim_M$. In order to compute a presentation for S it suffices to find the set of primitive elements of $\sim_M$. Thus, we must find the set of minimal nonnegative nontrivial integer solutions of the system of equations:

$$x_1 + x_3 + 2x_4 - x_5 - x_7 - 2x_8 = 0,$$

$$2x_1 + 3x_2 + x_3 + 3x_4 - 2x_5 - 3x_6 - x_7 - 3x_8 = 0.$$

By using an implementation of the algorithm appearing in [4] performed by our student P. Rodríguez Archilla, we obtain that this set of minimal solutions is

$$\{(0,0,0,1,0,0,0,1),(0,0,0,3,0,1,6,0),(0,0,0,1,1,0,1,0),(0,0,1,0,0,0,1,0),$$
$$(0,1,6,0,0,0,0,3),(0,1,0,0,0,1,0,0),(0,1,5,0,1,0,0,2),(0,1,0,3,6,0,0,0),$$
$$(0,1,1,2,5,0,0,0),(0,1,4,0,2,0,0,1),(0,1,2,1,4,0,0,0),(0,1,3,0,3,0,0,0),$$
$$(1,0,1,0,0,0,0,1),(1,0,0,2,0,1,5,0),(6,0,0,0,0,1,0,3),(5,0,0,0,0,1,1,2),$$
$$(2,0,0,1,0,1,4,0),(4,0,0,0,0,1,2,1),(3,0,0,0,0,1,3,0),(1,0,0,0,1,0,0,0)\}.$$

Whenever (a, b) is in a system of generators of a congruence, we do not need (b, a). Also we can remove elements of the form (a, a). Hence we obtain that

$$\{((0,0,0,3),(0,1,6,0)),((0,0,0,1),(1,0,1,0)),((0,1,5,0),(1,0,0,2)),$$
$$((0,1,0,3),(6,0,0,0)),((0,1,1,2),(5,0,0,0)),((0,1,4,0),(2,0,0,1)),$$
$$((0,1,2,1),(4,0,0,0)),((0,1,3,0),(3,0,0,0))\}$$

is a presentation (though not minimal) of S.

Example 7. Let S be the submonoid of $\mathbb{Z}_3 \times \mathbb{Z}$ spanned by

$$\{(\overline{1}, 2), (\overline{2}, 5)\}.$$

The equations of M are

$$x_1 + 2x_2 \equiv 0 \,(\mathrm{mod}\, 3),$$

$$2x_1 + 5x_2 = 0.$$

We already know that S is isomorphic to $\mathbb{N}^2/\sim_M$ and that for computing a presentation of S, we must determine the set $\mathfrak{P}(\sim_M)$. As we have seen before, we must find the set of minimal nonnegative nontrivial integer solutions of the system of equations:

$$x_1 + 2x_2 - x_3 - 2x_4 \equiv 0 \,(\mathrm{mod}\, 3),$$

$$2x_1 + 5x_2 - 2x_3 - 5x_4 = 0.$$

From the results appearing in [16], we can first compute the set of minimal nonnegative nontrivial integer solutions of the system of equations

$$x_1 + 2x_2 - x_3 - 2x_4 + 3x_5 - 3x_6 = 0,$$

$$2x_1 + 5x_2 - 2x_3 - 5x_4 = 0,$$

and then project onto the first four coordinates. By using once more the algorithm presented in [4] we obtain that this set is

$$\{(0,0,0,0,1,1),(0,1,0,1,0,0),(0,6,15,0,1,0),(15,0,0,6,0,1),(1,0,1,0,0,0)\}.$$

Therefore the set of minimal solutions of the original system of equations is

$$\{(0,1,0,1),(0,6,15,0),(15,0,0,6),(1,0,1,0)\},$$

which means that

$$\{((0,6),(15,0))\}$$

is a presentation of S.

3. Computing the irreducibles, an alternative approach

For $a=(a_1,\dots,a_p), b=(b_1,\dots,b_p)\in\mathbb{N}^p$ write

$$\gcd(a,b)=(\min\{a_1,b_1\},\dots,\min\{a_p,b_p\}),$$

$$\tau(a,b)=(a,b)-(\gcd(a,b),\gcd(a,b))$$

and

$$\operatorname{supp}(a)=\{i\in\{1,\dots,p\}\mid a_i\neq 0\}.$$

Clearly $\operatorname{supp}(a)\cap\operatorname{supp}(b)$ is empty if and only if $\gcd(a,b)=0$. Let R be a submonoid of $(\mathbb{N}^n\times\mathbb{N}^n,+)$. We say that *$R$ is simplified* if $\tau(R)\subseteq R$. We characterize this property and show its important connection with the computation of the set of primitive elements of a congruence of the form $\sim_M$.

Lemma 8. *Let $(a_1,b_1),\dots,(a_r,b_r)\in\mathbb{N}^n\times\mathbb{N}^n$ be such that $\gcd(a_i,b_i)=0$ for all $i\in\{1,\dots,r\}$. Let $(a,b)=(a_1,b_1)+\cdots+(a_r,b_r)$. If $\tau(a,b)\neq(a,b)$, then there exist $i,j\in\{1,\dots,r\}, i\neq j$, such that $\tau((a_i,b_i)+(a_j,b_j))\neq(a_i,b_i)+(a_j,b_j)$.*

Proof. If $\tau(a,b)\neq(a,b)$, then $\gcd(a,b)\neq 0$. Hence, there exists $k\in\operatorname{supp}(a)\cap\operatorname{supp}(b)=\operatorname{supp}(a_1+\cdots+a_r)\cap\operatorname{supp}(b_1+\cdots+b_r)$. Thus there are $i\in\{1,\dots,r\}$ such that $k\in\operatorname{supp}(a_i)$ and $j\in\{1,\dots,r\}$ with $k\in\operatorname{supp}(b_j)$. Since $\operatorname{supp}(a_i)\cap\operatorname{supp}(b_i)=\emptyset$, i cannot be equal to j. We deduce then that $\operatorname{supp}(a_i+a_j)\cap\operatorname{supp}(b_i+b_j)$ is not empty, and thus $\tau((a_i,b_i)+(a_j,b_j))\neq(a_i,b_i)+(a_j,b_j)$. □

Theorem 9. *Let R be a submonoid of $\mathbb{N}^n\times\mathbb{N}^n$ generated by $\{(a_1,b_1),\dots,(a_t,b_t)\}$. Assume that $\gcd(a_i,b_i)=0$ for all $i\in\{1,\dots,t\}$. Then R is simplified if and only if for all $i,j\in\{1,\dots,t\}$ with $i<j$ we have $\tau((a_i,b_i)+(a_j,b_j))\in R$.*

Proof.

Necessity: Trivial.

Sufficiency: Assume that R is not simplified. Let $(a,b)\in\mathbb{N}^n\times\mathbb{N}^n$ be minimal with respect to the condition that $(a,b)\in R$ and $\tau(a,b)\notin R$. There exist $i_1,\dots,i_r\in\{1,\dots t\}$ such that $(a,b)=(a_{i_1},b_{i_1})+\cdots+(a_{i_r},b_{i_r})$. Since $\tau(a,b)\neq(a,b)$, we have some $j,k\in\{1,\dots,r\}$ such that $\tau((a_{i_j},b_{i_j})+(a_{i_k},b_{i_k}))\neq(a_{i_j},b_{i_j})+(a_{i_k},b_{i_k})$ by Lemma 8. Hence $\tau((a_{i_j},b_{i_j})+(a_{i_k},b_{i_k}))=(a_{i_j},b_{i_j})+(a_{i_k},b_{i_k})-(c,c)$ for some $c\in\mathbb{N}^n\setminus\{0\}$. By hypothesis $\tau((a_{i_j},b_{i_j})+(a_{i_k},b_{i_k}))\in R$, and thus $(a,b)-(c,c)$ also belongs to R. But $(a,b)-(c,c)<(a,b)$ and $\tau((a,b)-(c,c))=\tau(a,b)\notin R$, contradicting the minimality of (a,b). □

Let M be a subgroup of $\mathbb{Z}^n$. Set, as we did above,

$$\sim_M=\{(a,b)\in\mathbb{N}^n\times\mathbb{N}^n\mid a-b\in M\}.$$

We already know that $\sim_M$ is a congruence on $\mathbb{N}^p$ and a submonoid of $\mathbb{N}^n\times\mathbb{N}^n$ generated by its irreducibles. It is well known (see for instance [17, Chapter 8]) that $\mathfrak{I}(\sim_M)=\mathfrak{P}(\sim_M)\cup\{(e_1,e_1),\dots,(e_n,e_n)\}$, where e_i is the ith row

of the n by n identity matrix. Clearly $\sim_M$ is simplified, moreover, if (a, b) and (c, d) are elements in $\sim_M$ such that $(a, b) \leqslant (c, d)$, then $(c, d) - (a, b)$ belongs also to $\sim_M$ (this is in fact the idea used to prove that $\sim_M$ is finitely generated as a monoid).

Let X be a subset of a monoid T. Then

$$\langle X\rangle = \{a_1x_1 + \cdots + a_kx_k \mid k \in \mathbb{N}, a_1, \ldots, a_k \in \mathbb{N}, x_1, \ldots, x_k \in X\}$$

is a submonoid of T, the submonoid of T generated by X.

Lemma 10. *Let $A \subseteq \sim_M$. Then $\mathfrak{P}(\sim_M) \subseteq A$ if and only if $\{(a, b) \in \sim_M \mid \gcd(a, b) = 0\} \subseteq \langle A\rangle$.*

Proof. Observe that every primitive element (a, b) satisfies $\gcd(a, b) = 0$. The necessity condition is trivial. For sufficiency, observe that every primitive element (a, b) of $\sim_M$ is in $\langle A\rangle$. By the minimality of (a, b), it must belong to A. □

For $u \in M$, one clearly has that $(u^+, u^-) \in \sim_M$ and that $\gcd(u^+, u^-) = 0$.

Proposition 11. *Let $A = \{(a_1, b_1), \ldots, (a_t, b_t)\}$ be a subset of $\sim_M$. Then $\mathfrak{P}(\sim_M) \subseteq A$ if and only if*

(1) *$\{a_1 - b_1, \ldots, a_t - b_t\}$ generates M as a monoid and*
(2) *$\langle A\rangle$ is simplificable.*

Proof.
Necessity: Assume that $\mathfrak{P}(\sim_M) \subseteq A$.

(1) Let $u \in M\backslash\{0\}$. Then $(u^+, u^-) \in \sim_M$. As $\mathfrak{I}(\sim_M)$ generates $\sim_M$ as a monoid, there exist $(a_{i_1}, b_{i_1}), \ldots, (a_{i_r}, b_{i_r}) \in \mathfrak{I}(\sim_M)$ such that $(u^+, u^-) = (a_{i_1}, b_{i_1}) + \cdots + (a_{i_r}, b_{i_r})$. Since $\operatorname{supp}(u^+) \cap \operatorname{supp}(u^-)$ is empty, every $(a_{i_j}, b_{i_j}) \in \mathfrak{P}(\sim_M)$ and thus $(a_{i_j}, b_{i_j}) \in A$. Hence, $u = u^+ - u^- = (a_{i_1} - b_{i_1}) + \cdots + (a_{i_r} - b_{i_r})$.
(2) Let $(a, b) \in \langle A\rangle$. Then $(a, b) \in \sim_M$ and $\tau(a, b) \in \sim_M$. By Lemma 10, we have that $\tau(a, b) \in \langle A\rangle$.

Sufficiency: By Lemma 10, it suffices to show that if $(a, b) \in \sim_M$ and $\gcd(a, b) = 0$, then $(a, b) \in \langle A\rangle$. As $a - b \in M$, there are $i_1, \ldots, i_r \in \{1, \ldots, t\}$ such that $a - b = a_{i_1} - b_{i_1} + \cdots + a_{i_r} - b_{i_r}$. Then there exists $c \in \mathbb{N}^n$ with $(a, b) + (c, c) = (a_{i_1}, b_{i_1}) + \cdots + (a_{i_r}, b_{i_r}) \in \langle A\rangle$. By (2), we have that $(a, b) = \tau((a, b) + (c, c)) \in \langle A\rangle$. □

Remark 12. Observe that if $\{v_1, \ldots, v_r\}$ is a system of generators of M as a group, then $\{v_1, -v_1, \ldots, v_r, -v_r\}$ generates M as a monoid.

Let A be a subset of $\mathbb{N}^n \times \mathbb{N}^n$. Denote the set of minimal elements with respect to the usual partial order $\leqslant$ by $\operatorname{Minimals}_{\leqslant}(A)$. The set A is *reduced* if $\operatorname{Minimals}_{\leqslant} A = A$.

Corollary 13. *Assume that $M \neq 0$. Let $A = \{(a_1, b_1), \ldots, (a_t, b_t)\}$. Then $\mathfrak{P}(\sim_M) = A$ if and only if*

(1) *$\{a_1 - b_1, \ldots, a_t - b_t\}$ generates M as a monoid,*
(2) *$\langle A\rangle$ is simplified,*
(3) *$\gcd(a_i, b_i) = 0$ for all $i \in \{1, \ldots, t\}$ and*
(4) *A is reduced.*

Proof. The necessity follows directly from our preceding results. For the converse, we already know (by Proposition 11) that $\mathfrak{P}(\sim_M) \subseteq A$. Assume that there exists $(a, b) \in A\backslash\mathfrak{P}(\sim_M)$. Then there exists $(c, d) \in \mathfrak{P}(\sim_M)$ such that $(c, d) < (a, b)$, contradicting the fact that A is reduced. □

Algorithm 14. Reduce(A).
INPUT: *$A \subseteq \mathbb{N}^n \times \mathbb{N}^n$ such that $\gcd(a, b) = 0$ for all $(a, b) \in A$ and $\{a - b \mid (a, b) \in A\}$ generates M as a monoid.*

OUTPUT: $B \subseteq \mathbb{N}^n \times \mathbb{N}^n$ *reduced such that* $\gcd(a,b) = 0$ *for all* $(a,b) \in B$ *and* $\{a-b \mid (a,b) \in B\}$ *generates* M *as a monoid.*

While $A \neq \text{Minimals}_{\leqslant}(A)$ *do*
$A = \text{Minimals}_{\leqslant}(A) \cup \{(a,b)-(c,d) \mid (a,b),(c,d) \in A \text{ and } (c,d) < (a,b)\}$.
Return A.

Let A_i be the resulting set after the ith execution of the while loop in the above algorithm ($A_0 = A$). Clearly, this gives a sequence

$$A_0 + (\mathbb{N}^n \times \mathbb{N}^n) \subseteq A_1 + (\mathbb{N}^n \times \mathbb{N}^n) \subseteq \cdots \subseteq A_i + (\mathbb{N}^n \times \mathbb{N}^n) \subseteq \cdots.$$

The algorithm stops, since this sequence must be stationary (this is an easy consequence of Dickson's Lemma; see for instance [17, Lemma 6.9]). The following proposition yields a procedure to compute $\mathfrak{P}(\sim_M)$ from a system of generators of M as a monoid (thus it suffices to know a basis of M).

Proposition 15. *Let* $A = \{(a_1,b_1),\ldots,(a_t,b_t)\} \subseteq \mathbb{N}^n \times \mathbb{N}^n$ *be such that*

(1) $\{a_1-b_1,\ldots,a_t-b_t\}$ *generates* M *as a monoid,*
(2) $\gcd(a_i,b_i) = 0$ *for all* $i \in \{1,\ldots,t\}$ *and*
(3) A *is reduced.*

Let $B = \{\tau((a_i,b_i)+(a_j,b_j)) \mid i,j \in \{1,\ldots,t\}, i<j\}$. *Then* $A = \mathfrak{P}(\sim_M)$ *if and only if* $\text{Reduce}(A \cup B) = A$.

Proof. If $A = \mathfrak{P}(\sim_M)$, then clearly $\text{Reduce}(A \cup B) = A$. Now assume that $\text{Reduce}(A \cup B) = A$. We prove that $A = \mathfrak{P}(\sim_M)$. In view of Corollary 13, it suffices to show that $\langle A \rangle$ is simplified. By Theorem 9, it suffices to show that $B \subseteq \langle A \rangle$. Let $(x,y) \in B$. Since $\text{Reduce}(A \cup B) = A$, there exists $i_1 \in \{1,\ldots,t\}$ such that $(a_{i_1},b_{i_1}) \leqslant (x,y)$. If equality holds, then $(x,y) \in A$. Otherwise, by using again that $\text{Reduce}(A \cup B) = A$, we have that there exists $i_2 \in \{1,\ldots,t\}$ such that $(a_{i_2},b_{i_2}) \leqslant (x-a_{i_1}, y-b_{i_1})$. If equality holds, then $(x,y) = (a_{i_1},b_{i_1}) + (a_{i_2},b_{i_2}) \in \langle A \rangle$. If this is not the case, we find i_3. If we continue doing this, we construct a descending chain that must become stationary. Thus, there exists $i_k \in \{1,\ldots,t\}$ such that $(a_{i_k},b_{i_k}) = (x-a_{i_1}-\cdots-a_{i_{k-1}}, y-b_{i_1}-\cdots-b_{i_{k-1}})$. Hence, $(x,y) \in \langle A \rangle$. □

Algorithm 16. *An algorithm to compute* $\mathfrak{P}(\sim_M)$.
INPUT: $A \subseteq \mathbb{N}^n \times \mathbb{N}^n$ *such that* $\gcd(a,b) = 0$ *for all* $(a,b) \in A$ *and* $\{a-b \mid (a,b) \in A\}$ *generates* M *as a monoid.*
OUTPUT: $\mathfrak{P}(\sim_M)$.

(1) $B = \text{Reduce}(A)$.
(2) $C = \{\tau((a,b)+(c,d)) \mid (a,b),(c,d) \in B \text{ and } (a,b) \neq (c,d)\}$.
(3) $D = \text{Reduce}(B \cup C)$.
(4) *If* $D = B$, *then return* B, *else set* $B = D$ *and goto* (2).

If M is given by generators, then it is easy to obtain A. If M is given in terms of equations, then we use the computation of the Smith normal form associated to the matrix of coefficients of these equations in order to compute a system of generators of M, as explained in [17, Chapter 2].

From Proposition 15, we deduce that if the algorithm returns something, then the output is precisely $\mathfrak{P}(\sim_M)$. Besides, the algorithm must stop for the same reason Algorithm 14 does: there are not infinite ascending chains of ideals in $\mathbb{N}^n \times \mathbb{N}^n$.

Lemma 17. *Under the standing hypothesis and definitions of Algorithm* 16, *if* $D \neq B$, *then* $B + (\mathbb{N}^n \times \mathbb{N}^n) \subsetneq D + (\mathbb{N}^n \times \mathbb{N}^n)$.

Proof. Observe that if (a,b) is not a minimal element of B, and $(c,d) \in B$ is such that $(c,d) < (a,b)$, then $(a,b) \in (a-c,b-d) + (\mathbb{N}^n \times \mathbb{N}^n)$; and precisely (a,b) is substituted by $(a-c,b-d)$ in the *Reduce* step. This proves that

$B + (\mathbb{N}^n \times \mathbb{N}^n) \subseteq D + (\mathbb{N}^n \times \mathbb{N}^n)$. The inclusion is proper because $D \neq B$ implies that new elements that are not greater than or equal to any element in B are added. □

3.1. A not so neat but faster version of the algorithm

The *Reduce* step in Algorithm 16 is highly time consuming. In this section we show an alternative way to avoid reduction in each loop.

For $x \in \mathbb{N}^n \times \mathbb{N}^n$ and L a sequence of elements of $\mathbb{N}^n \times \mathbb{N}^n$, we say that x is in *normal form* with respect to L if there is no element in L less than or equal to x (with respect to the usual partial order). If x is not in normal form with respect to L, then we can consider the first element of L less than or equal to x, say y_1. Then we can see whether $x - y_1$ is in normal form with respect to L. If not, we find $y_2 \in L$ such that $x - y_1 - y_2 \in \mathbb{N}^n \times \mathbb{N}^n$. Since there are only finitely many elements less than or equal to x, after a finite number of steps, we find $y_k \in L$ such that $x' = x - y_1 - \cdots - y_k$ is in normal form with respect to L. We say that x' is a normal form of x with respect to L. The following algorithm computes a normal form of x with respect to L.

Algorithm 18. NormalForm(x, L).
INPUT: $x \in \mathbb{N}^n \times \mathbb{N}^n$, *$L$ a sequence of elements in* $\mathbb{N}^n \times \mathbb{N}^n$.
OUTPUT: *a normal form of x with respect to L.*

Set $x' = x$.
While $M = \{y \in L \mid y \leqslant x'\} \neq \emptyset$ *do*
    $y = first(M)$,
    $x' = x' - y$.
Return x'.

Let B be any (finite) subset of $\mathbb{N}^n \times \mathbb{N}^n$. Set

- $B_0 = B$,
- $B_{i+1} = B_i \cup \{\text{NormalForm}(\tau((a, b) + (c, d)), B_i) \mid (a, b), (c, d) \in B_i\}$.

Let $x_i \in B_{i+1} \backslash B_i$. Then x_i is in normal form with respect to B_i, and thus $x_i \notin B_i + (\mathbb{N}^n \times \mathbb{N}^n)$. This proves that if $B_i \subsetneq B_{i+1}$, then $B_i + (\mathbb{N}^n \times \mathbb{N}^n) \subsetneq B_{i+1} + (\mathbb{N}^n \times \mathbb{N}^n)$. As pointed out above, there exists $k \in \mathbb{N}$ such that $B_k = B_{k+1}$ (see [17, Lemma 6.9]). Thus, the following algorithm stops after a finite number of execution steps. We say that $\bigcup_{i \geqslant 0} B_i = B_k$ is the *saturation* of B.

Algorithm 19. Saturation(B).
INPUT: $B \subseteq \mathbb{N}^n \times \mathbb{N}^n$
OUTPUT: $\bigcup_{i \geqslant 0} B_i$.

(1) $C = \{\text{NormalForm}(\tau((a, b) + (c, d)), B) \mid (a, b), (c, d) \in B, (a, b) \neq (c, d)\} \backslash \{0\}$.
(2) $D = B \cup C$.
(3) *If* $D = B$, *then return* B, *else set* $B = D$ *and goto* (1).

Lemma 20. *Let B be a finite subset of $\mathbb{N}^n \times \mathbb{N}^n$ with $\tau(B) = B$. Let $\bar{B} = \text{Saturation}(B)$. Then $R = \langle \bar{B} \rangle$ is simplified.*

Proof. Assume that $\bar{B} = \{(a_1, b_1), \ldots, (a_t, b_t)\}$. We use Theorem 9 to prove that R is simplified. Observe that

(1) For $(a, b) \in \mathbb{N}^n \times \mathbb{N}^n$, if $\text{NormalForm}((a, b), \bar{B}) = 0$, then $(a, b) \in R$.
(2) Since $\bar{B} = \text{Saturation}(B)$, for every $(a, b), (c, d) \in \bar{B}$, $\text{NormalForm}(\tau((a, b) + (c, d)), \bar{B}) = 0$.

The proof now follows easily. □

Lemma 21. *Let M be a subgroup of $\mathbb{Z}^n$, and let $\{b_1, \ldots, b_t\}$ be a basis of M. Let $B = \{(b_1^+, b_1^-), \ldots, (b_t^+, b_t^-)\}$, and let $\bar{B} = \text{Saturation}(B)$. If $(u^+, u^-) \in \sim_M$ is a primitive element such that $u = \sum_{i=1}^t a_i b_i$ with $a_1, \ldots, a_t \in \mathbb{N}$, then $(u^+, u^-) \in \bar{B}$.*

Proof. Since $u = \sum_{i=1}^t a_i b_i$, there exists $a \in \mathbb{N}^n$ such that $(u^+, u^-) + (a, a) = \sum_{i=1}^t a_i (b_i^+, b_i^-)$. This implies that $(u^+, u^-) + (a, a) \in \langle \bar{B} \rangle$. By Lemma 20, $\langle \bar{B} \rangle$ is simplified, and thus $(u^+, u^-) = \tau((u^+, u^-) + (a, a)) \in \langle \bar{B} \rangle$. Since (u^+, u^-) is irreducible, we deduce that $(u^+, u^-) \in \bar{B}$. □

Proposition 22. *Let M be a subgroup of $\mathbb{Z}^n$, and let $\{b_1, \ldots, b_t\}$ be a basis of M. For every element $s = (s_1, \ldots, s_t) \in \{-1, 1\}^t$, set*

$$B_s = \{((s_1 b_1)^+, (s_1 b_1)^-), \ldots, ((s_t b_t)^+, (s_t b_t)^-)\}.$$

Then

$$\mathfrak{P}(\sim_M) \subseteq \bigcup_{s \in \{-1,1\}^t} \text{Saturation}(B_s).$$

Proof. Let $(u^+, u^-) \in \mathfrak{P}(\sim_M)$. Then there exist $a_1, \ldots, a_t \in \mathbb{Z}$ such that $u = \sum_{i=1}^t a_i b_i$. For every $i \in \{1, \ldots, t\}$, if $a_i < 0$, then set $c_i = -a_i$, and let $s_i = -1$; otherwise, set $c_i = a_i$ and $s_i = 1$. Then, $u = \sum_{i=1}^t c_i (s_i b_i)$ with $c_1, \ldots, c_t \in \mathbb{N}$ and $\{s_1 b_1, \ldots, s_t b_t\}$ is a basis of M. By Lemma 21, $(u^+, u^-) \in \text{Saturation}(B_s)$. □

Thus, the following algorithm computes the set of primitive elements of $\sim_M$, for M a subgroup of $\mathbb{Z}^n$.

Algorithm 23. PrimitiveElements(M).
INPUT: *M a subgroup of $\mathbb{Z}^n$.*
OUTPUT: $\mathfrak{P}(\sim_M)$.

If a basis for M is not known, compute $\{b_1, \ldots, b_t\}$, a basis of M.
For every $s \in \{-1, 1\}^t$,
  compute $\overline{B_s} = \text{Saturation}(B_s)$, where B_s is defined as in Proposition 22.
Return $\text{Minimals}_{\leqslant} \bigcup_{s \in \{-1,1\}^t} \overline{B_s}$.

Observe that we can economize half of the work performed, since $(a, b) \in \mathfrak{P}(\sim_M)$ implies that $(b, a) \in \mathfrak{P}(\sim_M)$. Thus, we can consider s_1 to be always equal to one and add to the output the symmetric of the resulting set.

Example 24. We go back to Example 6. M is given by the equations

$$x_1 + x_3 + 2x_4 = 0,$$

$$2x_1 + 3x_2 + x_3 + 3x_4 = 0.$$

We identify (a, b) with $a - b$, since $\tau((a, b) + (c, d))$ in Step (1) of Algorithm 19 can be obtained directly from $(a - b) + (c - d)$. By taking $z = (a - b) + (c - d)$, we have that $\tau((a, b) + (c, d)) = (z^+, z^-)$.

A basis for M is $\{(-1, 0, -1, 1), (-3, 1, 3, 0)\}$. Thus, we must compute the saturation of $B_{(1,1)} = \{(-1, 0, -1, 1), (-3, 1, 3, 0)\}$ and $B_{(1,-1)} = \{(-1, 0, -1, 1), (3, -1, -3, 0)\}$. We outline this computation as follows.

- $(-1, 0, -1, 1) + (-3, 1, 3, 0) = (-4, 1, 2, 1)$, which is in normal form with respect to $B_{(1,1)}$. So we add it to $D := B_{(1,1)}$.
  Observe now that we do not have to check whether the normal form of $(-4, 1, 2, 1) + (-3, 1, 3, 0)$ with respect to $\{(-1, 0, -1, 1), (-3, 1, 3, 0), (-4, 1, 2, 1)\}$ is zero, since by looking at the signs we know that this is the case.
  $(-1, 0, -1, 1) + (-4, 1, 2, 1) = (-5, 1, 1, 2)$, which is in normal form, and we add it to our set D.
  $(-1, 0, -1, 1) + (-5, 1, 1, 2) = (-6, 1, 0, 3)$, also in normal form. Thus we include it in our set D.
  No more combinations are possible by using the argument of the signs given above (this actually speeds up the implementation of this procedure).

- $(-1, 0, -1, 1) + (3, -1, -3, 0) = (2, -1, -4, 1)$, in normal form; so we add it to $B_{(1,-1)}$.
  $(-1, 0, -1, 1) + (2, -1, -4, 1) = (1, -1, -5, 2)$ is also in normal form.
  $(-1, 0, -1, 1) + (1, -1, -5, 2) = (0, -1, -6, 3)$ in normal form as well, and there are no more possibilities.

Thus we obtain

$$((0, 0, 0, 1), (1, 0, 1, 0)), \quad ((0, 1, 3, 0), (3, 0, 0, 0)), \quad ((0, 1, 2, 1), (4, 0, 0, 0)),$$

$$((0, 1, 1, 2), (5, 0, 0, 0)), \quad ((0, 1, 0, 3), (6, 0, 0, 0)), \quad ((2, 0, 0, 1), (0, 1, 4, 0)),$$

$$((1, 0, 0, 2), (0, 1, 5, 0)) \quad \text{and} \quad ((0, 0, 0, 3), (0, 1, 6, 0)).$$

The rest of primitive elements are obtained by symmetry.

Example 25. We now revisit Example 7. The equations of M are

$$x_1 + 2x_2 \equiv 0 \,(\mathrm{mod}\, 3),$$

$$2x_1 + 5x_2 = 0,$$

and a basis for M is $\{(15, -6)\}$. Here computations are not necessary, since this set is already saturated (has only one element). Thus, the primitive elements are $((15, 0), (0, 6))$ and its symmetry $((0, 6), (15, 0))$.

The algorithms were implemented in GAP [23]. In the examples we run, Algorithm 23 was much faster than Algorithm 16. As we indicated above, the main reason for this is that the Reduce step is too slow. Algorithm 23 works fine if the rank of M is small, that is, if the number of independent defining equations approaches the number of generators of the monoid. This of course is a handicap of this method if we plan to use it for numerical semigroups, where there is only one equation. Next, we give a table comparing both algorithms. The second and third columns are execution times in milliseconds.

Semigroup	Algorithm 23	Algorithm 16
$\langle(3, 0, 0), (0, 3, 0), (0, 0, 3), (1, 2, 1), (1, 1, 2)\rangle$	0	15
$\langle(3, 0, 0), (0, 3, 0), (0, 0, 3), (1, 2, 1),$ $(2, 1, 2), (1, 1, 2)\rangle$	219	328
$\langle(4, 0, 0), (0, 3, 0), (0, 0, 2), (1, 2, 1),$ $(3, 1, 2), (1, 1, 2)\rangle$	735	9625
$\langle(3, 0, 0, 0), (0, 3, 0, 0), (0, 0, 3, 0), (0, 0, 0, 2),$ $(1, 2, 1, 1), (1, 1, 1, 2), (1, 1, 1, 1)\rangle$	453	3610

For numerical semigroups with a big minimal system of generators, if we are looking for a minimal presentation instead of the irreducibles, we recommend using the function `MinimalPresentationOfNumericalSemigroup` of the package [5] which implements the algorithm given in [15]. Observe that for computing some invariants of the semigroup, such as the elasticity, a minimal presentation is not enough (see [2]).

Semigroup	Algorithm 23	[5]
$\langle 12, 34, 57\rangle$	687	16
$\langle 91, 239, 372\rangle$	219	2594
$\langle 9, 15, 23, 37\rangle$	21687	15

In this table we highlight several aspects. For low embedding dimension (with respect to the number of minimal generators), the algorithm can somehow compete against the one given in [15] and implemented in [5]. Due to its programming structure, the latter algorithm gains some advantage when the multiplicity (with respect to the least minimal generator) is big. However, the last row and the remarks given above stress that the algorithm of [15] will work better for large embedding dimension.

References

[1] E. Briales, A. Campillo, C. Marijuán, P. Pisón, Minimal systems of generators for ideals of semigroups, J. Pure Appl. Algebra 124 (1998) 7–30.
[2] S.T. Chapman, J.I. García-García, P.A. García-Sánchez, J.C. Rosales, Computing the elasticity of a Krull monoid, Linear Algebra Appl. 336 (2001) 191–200.
[3] M. Clausen, A. Fortenbacher, Efficient solution of linear Diophantine equations, J. Symbolic Comput. 8 (1–2) (1989) 201–216.
[4] E. Contejean, H. Devie, An efficient incremental algorithm for solving systems of linear diophantine equations, Inform. and Comput. 113 (1994) 143–172.
[5] M. Delgado, P.A. García-Sánchez, J. Morais, numericalsgps: a GAP (The Gap Group. Gap—Groups, Algorithms, and Programming, Version 4.4, 2004 ⟨http://www.gap.system.org⟩) package on numerical semigroups ⟨http://www.gap-system.org/Packages/numericalsgps.html⟩.
[6] F. Di Biase, R. Urbanke, An algorithm to calculate the kernel of certain polynomial ring homomorphisms, Experimental Math. 4 (3) (1995) 227–234.
[7] E. Domenjoud, A.P. Tomás, From Elliott–MacMahon to an algorithm for general linear constraints on naturals, Principles and Practice of Constraint Programming—CP '95, Cassis, 1995, Lecture Notes in Computer Science, vol. 976, Springer, Berlin, 1995, pp. 18–35.
[8] J. Herzog, Generators and relations of abelian semigroups and semigroup rings, Manuscripta Math. 3 (1970) 175–193.
[9] P. Pisón, A. Vignerón Tenorio, Ideales de semigroups con torsión: cálculos mediante Maple V, EACA'96, Universidad de Sevilla, 1996.
[10] P. Pisón, A. Vignerón Tenorio, $\mathbb{N}$-solutions to linear systems over $\mathbb{Z}$, Linear Algebra Appl. 384 (2004) 135–154.
[11] L. Pottier, Bornes et algorithme de calcul des générateurs des solutions de systèmes diophantiens linéaires, C. R. Acad. Sci. Paris 311 (Série I) (1990) 813–816.
[12] L. Pottier, Minimal solutions of linear diophantine systems: bounds and algorithms, Rewriting Techniques and Applications, Como, 1991, Lecture Notes in Computer Science, vol. 488, Springer, Berlin, 1991, pp. 162–173.
[13] L. Rédei, The Theory of Finitely Generated Commutative Semigroups, Pergamon Press, Oxford, 1965.
[14] J.C. Rosales, On finitely generated submonoids of $\mathbb{N}^k$, Semigroup Forum 50 (1995) 251–262.
[15] J.C. Rosales, An algorithmic method to compute a minimal relation for any numerical semigroup, Internat. J. Algebra Comput. 6 (4) (1996) 441–455.
[16] J.C. Rosales, P.A. García-Sánchez, Nonnegative elements of subgroups of $\mathbb{N}^n$, Linear Algebra Appl. 270 (1998) 351–357.
[17] J.C. Rosales, P.A. García-Sánchez, Finitely Generated Commutative Monoids, Nova Science Publishers, New York, 1999.
[18] J.C. Rosales, P.A. García-Sánchez, J.I. García-García, Presentations of finitely generated submonoids of finitely generated commutative monoids, Internat. J. Algebra Comput. 12 (5) (2002) 659–670.
[19] J.C. Rosales, P.A. García-Sánchez, J.M. Urbano-Blanco, On presentations of commutative monoids, Internat. J. Algebra Comput. 9 (5) (1999) 539–553.
[20] J.C. Rosales, J.M. Urbano-Blanco, A deterministic algorithm to decide if a finitely presented abelian monoid is cancellative, Comm. Algebra 24 (13) (1996) 4217–4224.
[21] B. Sturmfels, Gröbner bases of toric varieties, Tôhoku Math. J. 43 (1991) 249–261.
[22] B. Sturmfels, Gröbner bases and convex polytopes, University Lecture Series, vol. 8, American Mathematical Society, Providence, RI, 1996.
[23] The GAP Group. GAP—Groups, Algorithms, and Programming, Version 4.4, 2004 ⟨http://www.gap-system.org⟩.