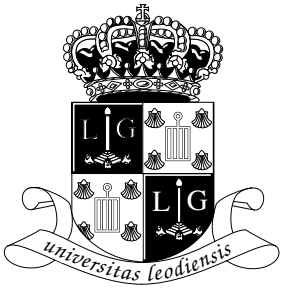
UNIVERSITE DE LIEGE



Faculté des Sciences Appliquées
Département d'électricité, électronique et informatique
Institut Montefiore

Presburger Arithmetic : From Automata to Formulas

Thèse présentée par

Louis Latour

en vue de l'obtention du titre
de Docteur en Sciences Appliquées

Année Académique 2005–2006

Abstract

Presburger arithmetic is the first-order theory of the integers with addition and ordering, but without multiplication. This theory is decidable and the sets it defines admit several different representations, including formulas, generators, and finite automata, the latter being the focus of this thesis. Finite-automata representations of Presburger sets work by encoding numbers as words and sets by automata-defined languages. With this representation, set operations are easily computable as automata operations, and minimized deterministic automata are a canonical representation of Presburger sets. However, automata-based representations are somewhat opaque and do not allow all operations to be performed efficiently. An ideal situation would be to be able to move easily between formula-based and automata-based representations but, while building an automaton from a formula is a well understood process, moving the other way is a much more difficult problem that has only attracted attention fairly recently.

The main results of this thesis are new algorithms for extracting information about Presburger-definable sets represented by finite automata. More precisely, we present algorithms that take as input a finite-automaton representing a Presburger definable set S and compute in polynomial time the affine hull over $\mathbb{Q}$ or over $\mathbb{Z}$ of the set S, i.e., the smallest set defined by a conjunction of linear equations (and congruence relations in $\mathbb{Z}$) which includes S. Also, we present an algorithm that takes as input a deterministic finite-automaton representing the integer elements of a polyhedron P and computes a quantifier-free formula corresponding to this set.

The algorithms rely on a very detailed analysis of the scheme used for encoding integer vectors and this analysis sheds light on some structural properties of finite-automata representing Presburger definable sets.

The algorithms presented have been implemented and the results are encouraging : automata with more than 100000 states are handled in seconds.

Acknowledgments

I would like to thank Pierre Wolper, for giving me the chance to join his research group and complete this thesis, and for his insightful supervision throughout my years in his group. It is also my pleasure to thank Bernard Boigelot for his countless explanations, from Presburger arithmetic and automata to C programming and single malt whisky. Thanks also to the other members of my jury, Pascal Gribomont, Michel Rigo, Charles Pecheur, Alain Finkel and Markus Müller-Olm, who have accepted to read and evaluate this thesis.

It is a good opportunity to thank my mother, for her lasting love, and my father, for his understanding with the relentless demands of graduate study. Thanks also to Isabelle, whose commitment in all she does will always be an inspiration. I also gratefully thank Roger for opening my mind.

Finally, I want to thank all those with whom I have shared good moments during those years. In particular, I thank Laurence, for her understanding and presence when I needed most, and Vincent, who always has a smile to share.

Contents

List of Figures

List of Symbols

$L_2 \div L_1$	left-quotient of L_1 with L_2
$\lvert a \rvert$	absolute value of number a, page 14
$\lvert S \rvert$	cardinality of the set S
$\langle w \rangle_r$	the number $z \in \mathbb{Z}$ such that w is an encoding in basis r of z when using the synchronous encoding scheme $E_{S(r)}$, page 56
$\langle w \rangle_{r,n}$	the vector $\mathbf{z} \in \mathbb{Z}^n$ such that w is an encoding in basis r of $\mathbf{z}$ when using the synchronous encoding scheme $E_{S(r)}$, page 58
$\exists_I(S)$	the projection of S with respect to the set I of components, page 62
$\exists_i(S)$	the projection of S with respect to the ith component, page 61
$\lceil a \rceil$	smallest integer larger or equal to a, page 14
$\lfloor a \rfloor$	largest integer smaller or equal to a, page 14
$\mathcal{A}$	finite automaton, page 41
ε	empty word, page 41
$a \equiv_m b$	integer a is equal to integer b modulo m, page 14
$w \div L$	the language $\{v \mid wv \in L\}$
$\mathbb{N}$	set of natural numbers, page 14
$\mathbb{Q}$	set of rational numbers, page 14
$\mathbb{R}$	set of real numbers, page 14

$\mathbb{Z}$	set of integer numbers, page 14
$\mathbb{Z}_m$	set of equivalence classes of the congruence modulo m relation, page 14
$\mathbf{A}, \mathbf{B}, \mathbf{C}$	matrices, page 15
$\mathbf{A}[i, j]$	entry located in ith row and jth row of matrix $\mathbf{A}$
$\mathbf{a}, \mathbf{b}, \mathbf{c}$	vectors, page 15
$\mathbf{a}[i]$	ith entry of vector $\mathbf{a}$, page 15
$\mathbf{A}^{\mathtt{t}}$	matrix $\mathbf{A}$ transposed, i.e. $\mathbf{A}^{\mathtt{t}}[i, j] = \mathbf{A}[j, i]$
$A \times B$	Cartesian product of two sets A and B, page 13
A, B, C	sets, page 13
a, b, c	scalars or words
$\mathrm{aff}_{\mathbb{D}}(S)$	affine hull over $\mathbb{D}$ of the set S, with $\mathbb{D} = \mathbb{Q}, \mathbb{Z}$ or $\mathbb{Z}_m$, page 23
$\mathbf{A}^{+}\mathbf{x} \leq \mathbf{b}^{+}$	subsystem of $\mathbf{Ax} \leq \mathbf{b}$ formed by the inequations which are not implicit equations, page 29
$\mathbf{A}^{=}\mathbf{x} \leq \mathbf{b}^{=}$	subsystem of $\mathbf{Ax} \leq \mathbf{b}$ formed by the implicit equations, page 29
$\text{char-cone}(P)$	characteristic cone of the polyhedron P
$\mathrm{cone}_{\mathbb{D}}(S)$	conic hull over $\mathbb{D}$ of the set S, with $\mathbb{D} = \mathbb{Q}, \mathbb{Z}$ or $\mathbb{Z}_m$, page 23
$\mathrm{conv}_{\mathbb{D}}(S)$	convex hull over $\mathbb{D}$ of the set S, with $\mathbb{D} = \mathbb{Q}, \mathbb{Z}$ or $\mathbb{Z}_m$, page 23
DFA	deterministic finite automaton, page 44
$E_{I(r)}$	synchronous interleaved encoding scheme, page 79
$E_{R(r)}$	reverse synchronous encoding scheme, page 74
$E_{S(r)}$	synchronous encoding scheme, page 58
FA	finite automaton, page 41
$\mathbf{I}_n$	identity matrix of dimension n, page 15

iff	if and only if
$L_{\mathcal{A}}(q)$	set of words labeling paths from state q to a final state in the FA $\mathcal{A}$, page 42
$L_{\mathcal{A}}(q_1 \rightarrow q_2)$	set of words labeling paths from state q_1 to state q_2 in the FA $\mathcal{A}$, page 42
$L(\mathcal{A})$	language accepted by the FA $\mathcal{A}$, page 42
$\mathrm{lin}_{\mathbb{D}}(S)$	linear hull over $\mathbb{D}$ of the set S, with $\mathbb{D} = \mathbb{Q}, \mathbb{Z}$ or $\mathbb{Z}_m$, page 23
$\log_r(a)$	logarithm in base r of a, page 14
M	the $\mathbb{Z}$-module such that $\mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}) = \mathbf{x} + M$ for some $\mathbf{x} \in S_{\mathcal{A}}$, page 102
$b \mod m$	the integer in $\{0, \ldots, m-1\}$ congruent modulo m to b, page 14
M_q	the $\mathbb{Z}$-module such that if $S^q_{\mathcal{A}} \neq \emptyset$, $\mathrm{aff}_{\mathbb{Z}}(S^q_{\mathcal{A}}) = \mathbf{x}_q + M_q$ for some $\mathbf{x}_q \in S^q_{\mathcal{A}}$, page 102
NDD	Number Decision Diagram, page 59
o	symbol $(0, \ldots, 0)$ of Σ_r^n, page 59
$\mathrm{pre}(w)$	set of prefixes of the word w, page 41
RVA	Real Vector Automaton
$S_{\mathcal{A}}$	the set of integer vectors represented by the NDD $\mathcal{A}$, page 59
$S^q_{\mathcal{A}}$	the set of vectors whose encodings label paths from any $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ to q in the NDD $\mathcal{A}$, page 59
SCC	strongly connected component, page 46
V_q	the vector space over $\mathbb{Q}$ such that $\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}} = \mathbf{x} + V_q$ for some $\mathbf{x} \in \mathbb{Q}^n$, page 92
V_q	the vector space over $\mathbb{Q}$ such that if $S^q_{\mathcal{A}} \neq \emptyset$, $\mathrm{aff}_{\mathbb{Q}}(S^q_{\mathcal{A}}) = \mathbf{x} + V_q$ for some $\mathbf{x} \in \mathbb{Q}^n$, page 92

Chapter 1

Introduction

1.1 Presburger Arithmetic and its Applications

Back to the work of Frege, formal logic [End01] has been the subject of intensive research and has lead to significant developments in many areas, including mathematics and computer science. In the context of Tarski's research project of the years 1926-1929, Mojżesz Presburger has introduced in [Pre29, Pre91] the first-order theory of the integer numbers with addition and ordering relation $\langle \mathbb{Z}, 0, 1, +, < \rangle$, called *Presburger arithmetic*. By defining a set of axioms and applying the general procedure of elimination of quantifiers, Presburger showed that this theory is *complete* and therefore *decidable*. This result is of particular significance in light of Godël's incompleteness proofs, and Church's and Rosser's undecidability results, directed toward classical first order theories powerful enough to represent the axioms of Peano arithmetic which involve only the constants 0 and 1, addition and multiplication.

Since Presburger arithmetic is expressive, and yet decidable, it is a useful tool for many problems. These include discrete optimization problems [Sch86], compiler optimization techniques [OME], program analysis tools [SKR98] and state-space exploration problems [Boi99, Ler03].

This work has been triggered by applications of Presburger arithmetic in the context of state-space exploration. State-space exploration refers to the computation of all possible states of computer systems. In many instances of computer systems, states can be represented by integer vectors, and both sets of states and the transition relation can be defined in Presburger arithmetic. The general issue is then to compute the set of reachable states by recursively applying the reflexive closure R of the transition relation on the set of initial states I, i.e. computing the

least fixpoint $R^*(I)$ of the sequence

$$I \subseteq R^{(1)}(I) \subseteq R^{(2)}(I) \subseteq R^{(3)}(I) \subseteq \ldots$$

where $R^{(i)}(X)$ represents $\underbrace{R(R(\ldots R}_{i}(I)))$.

Concretely, the general approach is the following. If $\varphi_0(\mathbf{x})$ and $\psi(\mathbf{x}, \mathbf{x}')$, where $\mathbf{x}$ and $\mathbf{x}'$ are integer vector variables, are Presburger formulas defining respectively the set of initial states I and the reflexive closure R of the transition relation, the set of states reachable in at most one step is defined by the formula

$$\varphi_1(\mathbf{x}) =_{def} \exists \mathbf{y} \left(\varphi_0(\mathbf{y}) \wedge \psi(\mathbf{y}, \mathbf{x})\right).$$

The set of states reachable in at most $k+1$ steps is defined by the formula

$$\varphi_{k+1}(\mathbf{x}) =_{def} \exists \mathbf{y} \left(\varphi_k(\mathbf{y}) \wedge \psi(\mathbf{y}, \mathbf{x})\right).$$

The fixpoint is reached after k steps if k is the smallest integer such that

$$R^{(k+1)}(I) \subseteq R^{(k)}(I).$$

This inclusion holds if and only if the following first-order formula holds.

$$\forall \mathbf{x} \ \left(\varphi_{k+1}(\mathbf{x}) \Rightarrow \varphi_k(\mathbf{x})\right).$$

Note that a fixpoint does not always exist and that there exist methods for accelerating the computation of the fixpoint (see [Boi99, Ler03]).

1.2 Representing Presburger Definable Sets

1.2.1 Formulas

The most immediate way of handling Presburger definable sets, i.e. sets definable in Presburger arithmetic, is to work directly with the formulas. An important issue is to be able to check the satisfiability of a formula, i.e. the existence of a solution satisfying a given formula $\varphi(x_1, \ldots, x_n)$, where x_1, …, x_n are the free variables. The general procedure for this problem relies on a quantifier-elimination method. However, Presburger arithmetic as such does not admit the elimination of quantifiers. For example, there is no quantifier-free formula equivalent to the formula $\exists y \ (x = 2 \cdot y)$. We can overcome this by adding new symbols $\equiv_m$, with

$m \in \{1, 2, \ldots\}$, for congruence relations modulo m, i.e. $x \equiv_m y$ if and only if there exists an integer k such that $m \cdot k = x - y$, where $m \cdot k$ is simply an abbreviation of $\underbrace{k + \ldots + k}_{m}$.

The general procedure for testing the satisfiability of a formula $\varphi(x_1, \ldots, x_n)$ is the following.

- Generate a new formula ψ by quantifying existentially all free variables, i.e. $\psi =_{def} \exists x_1 \exists x_2 \ldots \exists x_n \; \varphi(x_1, \ldots, x_n)$.
- Generate a quantifier-free formula ψ' equivalent to ψ.
- Decide whether ψ' holds.

Deciding whether the quantifier-free formula ψ' holds is simple since there are no variable in ψ', i.e. ψ' is a Boolean combination of formulas $a \equiv_m b$, $a \leq b$ or $a = b$, where $a, b, m \in \mathbb{Z}$ with $m \geq 1$. So, the non-trivial operation is the elimination of the quantifiers. A detailed procedure is given in Section 2.5.

The explicit handling of formulas and the quantifier elimination procedure have been successfully implemented e.g. in the Omega package [OME]. The major drawback of handling explicit formulas is the lack of *canonicity*. Indeed, there exist generally many formulas corresponding to the set, and there is no criterion that favors one particular formula rather than the others. As a result, if a Presburger set is built incrementally, the final formula can be large although the represented set is simple. In addition, if the construction is a fixpoint computation, each step requires a test of inclusion, which is in the worst-case triply exponential in the length of the formula [Opp78].

1.2.2 Generators

Presburger sets can be represented by means of generators according to the characterization of Presburger-definable sets as semi-linear sets [GS66]. A subset S of $\mathbb{Z}^n$ is *linear* if there exist vectors $\mathbf{c}, \mathbf{p}_1, \ldots, \mathbf{p}_k \in \mathbb{Z}^n$ such that

$$S = \{\mathbf{c} + \sum_{i=1}^{k} a_i \cdot \mathbf{p}_i \mid a_1, \ldots, a_k \in \mathbb{N}\}.$$

In the following, given the vector $\mathbf{c} \in \mathbb{Z}^n$ and the finite set $P \subseteq \mathbb{Z}^n$, we use the notation $(\mathbf{c}; P)$ to denote the linear set $\{\mathbf{c} + \sum_{i=1}^{k} a_i \cdot \mathbf{p}_i \mid \mathbf{p}_i \in P \wedge a_i \in \mathbb{N}\}$.

A subset S of $\mathbb{Z}^n$ is *semi-linear* if it is a finite union of linear sets, i.e. if there exists a finite set of vectors $\mathbf{c}_1, \ldots, \mathbf{c}_t \in \mathbb{Z}^n$ and a finite set of finite sets $P_1, \ldots, P_t \subseteq \mathbb{Z}^n$ such that $S = \bigcup_{j\in\{1,\ldots,t\}}(\mathbf{c}_j; P_j)$. It has been shown in [GS66] that a subset of $\mathbb{Z}^n$ is a semi-linear set if and only if there exists a a Presburger formula defining the set and the conversion from one representation to the other is computable[1].

Clearly, given a semi-linear set $S = \cup_{i\in\{1,\ldots,k\}}(\mathbf{c}_i; P_i)$, there exists a Presburger formula defining S. For the converse implication, one notes first that for each set S corresponding to the integer elements of a (convex) polyhedron, i.e. the integer solutions of a system of linear inequations, there exists a semi-linear set generating S. Also, applying any operation corresponding to a Boolean operators or to an existential quantification on semi-linear sets produces another semi-linear set, whose semi-linear representation (i.e. the finite sets of generators) can be effectively computed from semi-linear representations of the initial sets. So, given a Presburger formula $\varphi(x_1, \ldots, x_n)$, in which atomic formulas are linear inequations, a semi-linear representation of the set defined by φ is built incrementally. One generates first semi-linear representations of sets corresponding to the linear inequations (see [AC97]), and then applies the Boolean operators and the existential quantification. As an example, a semi-linear representation of the set defined by the formula $x_1 \geq 0 \wedge x_2 \geq 0 \wedge \exists y\ (x_1 + x_2 - 2 \cdot y \leq 3 \wedge x_1 - 2 \cdot x_2 - 3 \cdot y \leq -1 \wedge -x_1 + y \leq 0)$ is

$$\bigcup_{\mathbf{c}\in\{(0,1),(0,2),(0,3),(1,0)\}} \big(\mathbf{c}; \{(1,0),(1,1)\}\big).$$

Clearly, given the semi-linear representation of a set, finding one element in the set is trivial. The costly operations are the computation of the semi-linear representation of the intersection between two semi-linear sets. Yet, another shortcoming is that in general, the semi-linear representations are not canonical. Nonetheless, this representation has been used in [RV02] in the context of sets restricted to be positive integer elements in finite union of (convex) polyhedra, i.e. sets which can be defined by Boolean combinations of inequations.

1.2.3 Finite Automata

A third approach for handling Presburger sets is to represent them via finite automata. The idea of representing sets of numbers with finite state machines dates

[1]The proof in [GS66] is done for subsets of $\mathbb{N}^n$ and Presburger arithmetic over the natural numbers, but it is easily generalized to subsets of $\mathbb{Z}^n$.

back to the work of Büchi [Buc60] and is the following. Any positive integer can be encoded as a finite word $w = d_l \dots d_0$ of digits belonging to the set $\Sigma_r = \{0, \dots, r-1\}$ such that

$$a = \sum_{i=0}^{l} d_i \cdot r^i.$$

The encoding of a is not unique since prefixing any encoding by 0 generate another encoding of the same number. This encoding scheme can be generalized for all integers by requiring that the encodings of $z \in \mathbb{Z}$ such that $-r^p \leq z < r^p$, where $p \geq 0$, have at least $p+1$ digits. If $z < 0$, then, the encodings of z are the last $p'+1$ digits of $r^{p'+1} + z$ for all $p' \geq p$. For example, the words 0120 and 2201 are 3-encodings of the numbers 15 and -8 respectively. Indeed, we have

$$\begin{aligned} -3^3 \leq 15 < 3^3 \quad &\text{and} \quad 15 = 3^2 + 2 \cdot 3^1 + 0 \cdot 3^0, \\ -3^2 \leq -8 < 3^2 \quad &\text{and} \quad 3^4 - 8 = 2 \cdot 3^3 + 2 \cdot 3^2 + 0 \cdot 3^1 + 1 \cdot 3^0. \end{aligned}$$

According to the above scheme, the first digit of the encodings of an integer z will be 0 if $z \geq 0$ and $r-1$ if $z < 0$. For this reason, the first digit of an encoding is called the *sign digit*. One notes that the encoding of a number is not unique since prefixing an encoding by its sign digit leads to another encoding of the same number. For instance, 201 and 2201 are both 3-encodings of -8. One generalizes this encoding scheme to vectors of integer numbers by reading simultaneously the digits of the r-encodings of the components, provided that they share the same length. An r-encoding of a vector $\mathbf{a} \in \mathbb{Z}^n$ is therefore a word over the alphabet Σ_r^n. For example, the word $(0,2)(1,2)(2,0)(0,1)$ is a 3-encoding of the vector $(15, -8)$. The restriction regarding the length of the encodings of the vector components is easily dealt with since one can always prefix a r-encoding by any sequence of sign digits without modifying the encoded number. This encoding scheme is further detailed in Section 5.1. We say that a set S of positive integer vectors is *r-recognizable* if there exists a finite automaton accepting the sets of all r-encodings of the elements in S.

Any Presburger-definable set is r-recognizable for any $r \geq 2$.

- First note that finite automata accepting the encodings of the elements in sets corresponding to the formulas $x = y$, $x \leq y$ and $x + y = z$ are given in Fig. 1.1, Fig. 1.2, and 1.3 respectively[2].

[2]Recall that the words accepted by a finite automaton are those labeling paths from the initial state (denoted as ) to a final state (denoted as ).

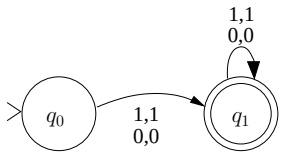


Figure 1.1: Finite automaton accepting the 2-encodings of the set $\{(x, y) \in \mathbb{Z}^2 \mid x = y\}$.

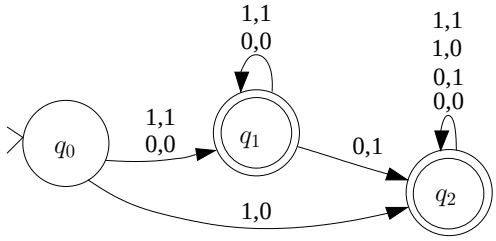


Figure 1.2: Finite automaton accepting the 2-encodings of the set $\{(x, y) \in \mathbb{Z}^2 \mid x \leq y\}$.

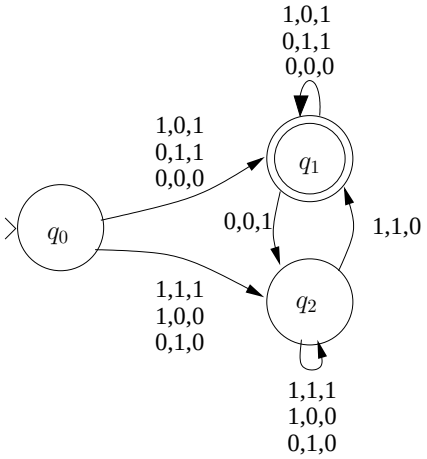


Figure 1.3: Finite automaton accepting the 2-encodings of the set $\{(x, y, z) \in \mathbb{Z}^3 \mid x + y = z\}$.

- Also, the sets of r-encodings of the union, intersection, Cartesian productand difference of two sets S_1, S_2 are the union, intersection, Cartesian product and the difference of the sets of r-encodings of the elements in S_1 and S_2. So, computing the union, intersection, Cartesian product and difference of sets can be done by applying the corresponding operations on automata.

- Similarly, the set of r-encodings of the complement of a set $S \subseteq \mathbb{Z}^n$ is the set of r-encodings which are not encodings of elements of S, and an automaton representing this set can be computed from the automaton representing S.

- Finally, given an automaton accepting the set of r-encodings of the elements of a set $S \subseteq \mathbb{Z}^n$, one obtains an automaton accepting the set of r-encodings of the set $\{(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \mid (x_1, \ldots, x_n) \in S\}$ by removing the ith component of the transition labels and ensuring that if some encoding of a vector is accepted in the automaton, then all the smaller encodings of the vectors are also accepted. This is required when the smallest encoding of the removed component of a vector is larger than the smallest encoding of the other component. For instance, all 2-encodings of the vector $(2, 0, 0)$ have at least 3 symbols whereas one encoding of the vector $(0, 0)$ has only one symbol.

- Since any Presburger-definable set can be defined by a formula obtained by combining atomic formulas of types $x + y = z$ and $x \leq y$ with Boolean operators and existential quantifiers, we conclude that any Presburger-definable set is r-recognizable.

For example, a finite automata accepting the 2-encodings of the elements in the set defined by the formula $\exists y\ (x_1 + x_2 - 2 \cdot y \leq 3 \wedge x_1 - 2 \cdot x_2 - 3 \cdot y \leq -1 \wedge -x_1 + y \leq 0)$ is given in Fig 1.4.

In 1969, A. Cobham proved that any subset of $\mathbb{N}$ r-recognizable for all $r \geq 2$ is definable in Presburger arithmetic [Cob69], and this result has been generalized by A. Semenov in [Sem77] for subsets of $\mathbb{N}^n$, i.e. any subset of $\mathbb{N}^n$ r-recognizable for all $r \geq 2$ is definable in Presburger arithmetic. Another proof of this result can be found in [BHMV94, Muc03]. The generalization to subsets of $\mathbb{Z}^n$ does not present any additional difficulties.

The automata-based representations of Presburger sets have been recently investigated in practical applications [WB95, BC96, Boi99]. This approach presents

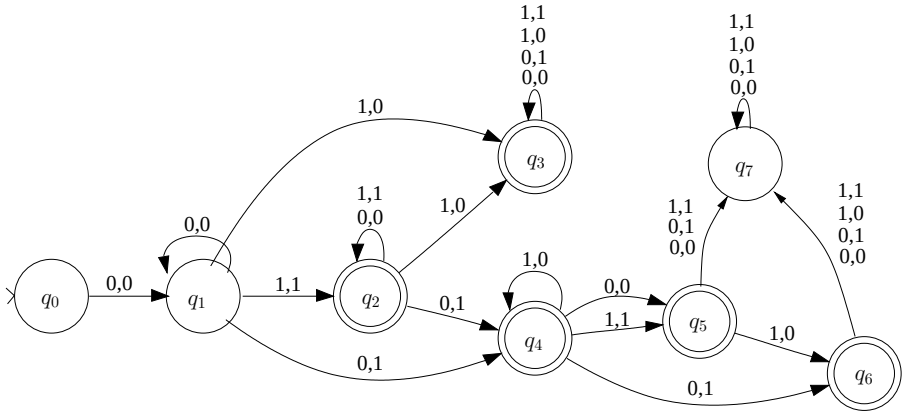


Figure 1.4: Finite automaton accepting the 2-encodings of the set $\{(x_1, x_2) \in \mathbb{N}^2 \mid \exists y\ (x_1 + x_2 - 2 \cdot y \leq 3 \wedge x_1 - 2 \cdot x_2 - 3 \cdot y \leq -1 \wedge -x_1 + y \leq 0)\}$.

two main advantages. First, automata have a canonical form. Second, finite automata theory has been investigated for a long time, and efficient (polynomial time complexity) procedures exist for set operations performed on automata in the canonical form (except for projection which may require an exponential time determinization in order to remain in canonical form). Those characteristics make the automata-based approach particularly suitable for applications involving many set manipulations. Still they are some drawbacks associated to automata-based representations. First, simple sets can lead to large automata. For example, the canonical automaton representing the set of integer solutions satisfying the linear equation $x - 10000y = 0$ has more than 10000 states. Secondly, some simple mathematical transformations such as an affine transformation are costly when performed on automata. Indeed, those operations are achieved by means of a sequence of product, intersection, projection and determinization operations, and the determinization can lead to an exponential increase of the number of states. Finally, automata are somewhat opaque representations and existing procedures may not be sufficient or efficient if one is interested in extracting the information contained in the automata. For example, in the context of state-space exploration, one would like to identify quickly the unexpected states or to use the results of an analysis performed with automata-based representations as input to tools using other representations, but those issues are not properly addressed by existing procedures.

1.3 Our Contribution

In this thesis, we address some of the above drawbacks related to automata-based representation of sets of integer vectors.

We present algorithms that take as input a finite automaton representing a Presburger definable set S and compute in polynomial time the affine hull over $\mathbb{Q}$ or over $\mathbb{Z}$ of the set S. The affine hull over $\mathbb{D}$, with $\mathbb{D} \in \{\mathbb{Q}, \mathbb{Z}\}$, of a set S, denoted $\mathrm{aff}_{\mathbb{D}}(S)$, is the smallest set containing all the affine combinations of S, i.e. if $\mathbf{x}_1, \ldots, \mathbf{x}_k \in S$, then

$$\sum_{i=1}^{k} a_i \cdot \mathbf{x}_i \in \mathrm{aff}_{\mathbb{D}}(S),$$

for all $a_1, \ldots, a_k \in \mathbb{D}$ such that $\sum_{i=1}^{k} a_i = 1$. Interestingly, given a set $S \subseteq \mathbb{Q}^n$, $\mathrm{aff}_{\mathbb{Q}}(S)$ is the set of elements (in $\mathbb{Q}^n$) satisfying a conjunction of linear equations and is the smallest set containing S with this property. Similarly, given a set $S \subseteq \mathbb{Z}^n$, $\mathrm{aff}_{\mathbb{Z}}(S)$ is the set of elements (in $\mathbb{Z}^n$) satisfying a conjunction of equations and congruence relations and is the smallest set containing S with this property.

We also present an algorithm that takes as input a deterministic finite automaton representing the integer elements of a polyhedron P and computes a quantifier-free formula corresponding to this set.

Our algorithms rely on a very detailed analysis of the scheme used for encoding integer vectors and this analysis sheds light on some structural properties of finite automata representing Presburger definable sets.

The general problem of computing formulas or generators corresponding to sets of integer vectors represented by automata has been addressed in [Ler03, Lat04, Lug04, Ler04b, Ler04a, Ler05, Lat05a, FL05]. What really distinguishes our approach is that our focus is on practical solutions. Indeed, the algorithms presented in this thesis have been implemented and applied to automata occurring in practical applications, with more than 100000 states. Also, the formulas generated by our algorithms present the advantage of being such that a polynomial time procedure exists for generating an automaton representing the same set.

1.4 Overview of the Thesis

The thesis is divided in two parts.

In the first part, we review the main theoretical concepts required for understanding our contribution. Chapter 2 recalls some basic notions regarding sets,

numbers and first-order theories. In Chapter 3, we detail some notions of algebra; those include the concept of (convex) polyhedron, vector space over $\mathbb{Q}$ and $\mathbb{Z}$-module and some of their properties. Relevant notions regarding finite automata are provided in Chapter 4, and in Chapter 5, we show how finite automata can represent sets of integer vectors.

In the second part, we present our main contributions. In Chapter 6, we present methods for computing over-approximations of sets represented by automata. More precisely, we present methods for computing the affine hulls over $\mathbb{Q}$ and over $\mathbb{Z}$ of the sets represented by automata.

In Chapter 7, we characterize automata representing integer elements of polyhedra, and present an algorithm which, given a finite automaton representing the integer elements of a polyhedron, generates both a formula whose integer solutions are the integer elements of the polyhedron as well as a semi-linear representation of this set.

Finally, in Chapter 8, we give general conclusions and present directions for future work.

Part I

Theoretical Background

Chapter 2

Preliminaries

2.1 Sets and Relations

We recall useful facts regarding sets and relations, and introduce some notations.

Given two sets A and B, we write $A \subseteq B$ if all elements of A are in B and $A \subset B$ if in addition there is at least one element in B which is not in A. Also, the cardinality of a finite set S is denoted by $|S|$.

A *binary operation $*$ on a set* is a rule which assigns to each pair of elements of the set an element of the set. A binary operation $*$ on a set S is *commutative* if $a * b = b * a$ for all $a, b \in S$. The operation is *associative* if $(a * b) * c = a * (b * c)$ for all $a, b, c \in S$.

The *Cartesian product* of two sets A and B, denoted $A \times B$, is the set of all ordered pairs (x, y) such that $x \in A$ and $y \in B$. The set A^n is the set of n-tuples of elements of A.

A *relation* R on a set S is a subset of S^2. A *n-ary relation on* A is a a subset of A^n.

We say that a binary relation R on set S is

1. *reflexive* if $(a, a) \in R$ for all $a \in S$;

2. *transitive* if $(a, b) \in R$ and $(b, c) \in R$ implies that $(a, c) \in R$;

3. *symmetric* if $(a, b) \in R$ implies $(b, a) \in R$;

A relation R that is reflexive, symmetric and transitive is said to be an *equivalence relation*.

Given a set S and an equivalence relation R on S, the *equivalence class* of an element $a \in S$, denoted $[a]_R$, is the subset of all elements in S which are equivalent to a :

$$[a]_R = \{b \in S | (a, b) \in R\}.$$

A *partition* of a set S is a set of non-empty subsets of S such that every element of S is in exactly one of these subsets.

If an equivalence relation R is given on the set S, then the set of all equivalence classes of R forms a partition of S. Conversely, if a partition P is given on S, we can define an equivalence relation R on S by writing $(a, b) \in R$ iff there exists a member of P which contains both a and b. The notions of *equivalence relation* and *partition* are thus essentially equivalent.

The number of equivalence classes generated by an equivalence relation R is the *index* of R. Given two equivalence relations R_1, R_2 defined on the same set S, R_1 is a *refinement* of R_2 if for each equivalence class $[a]_{R_1}$ of R_1, there exists an equivalence class $[b]_{R_2}$ of R_2 such that $[a]_{R_1} \subseteq [b]_{R_2}$.

A *(total) function* from X to Y is a subset f of the cartesian product $X \times Y$, such that for each $x \in X$, there is a unique $y \in Y$, denoted $f(x)$, such that the ordered pair (x, y) belongs to f.

A *partial function* from X to Y is a subset f of the cartesian product $X \times Y$, such that for each $x \in X$, there is at most one element $y \in Y$, denoted $f(x)$, such that the ordered pair (x, y) belongs to f. Given an element $x \in X$, if there is no element $y \in Y$ such that $(x, y) \in f$, one writes $f(x) = \perp$.

2.2 Numbers, Vectors, Matrices

As usual, $\mathbb{R}$, $\mathbb{Q}$, $\mathbb{Z}$ and $\mathbb{N}$ denote the sets of real, rational, integer and natural numbers.

Given a number a, $|a|$, $\lceil a \rceil$ and $\lfloor a \rfloor$ denote respectively the absolute value of a, the smallest integer larger or equal to a and the largest integer smaller or equal to a respectively. Also, $\log_r(a)$ is the logarithm in base r of a and is well defined if $a, r > 0$. The notation $\log a$ is also used for $\log_2(a)$.

Given an integer m with $m > 0$, two integers $a, b \in \mathbb{Z}$ are *congruent modulo* m, denoted $a \equiv_m b$, if m divides $a - b$. We use the notation $b \mod m$ to denote the integer in $\{0, \ldots, m-1\}$ congruent modulo m to b. Congruence modulo m is an equivalence relation, and the set of equivalence classes of this relation is denoted by $\mathbb{Z}_m$. In the following, each class in $\mathbb{Z}_m$ is represented by the element

a in the class such that $a \in \{0, \ldots, m-1\}$. So, any addition or multiplication of elements in $\mathbb{Z}_m$ corresponds to addition or multiplication in $\mathbb{Z}$ modulo m so that the result is in $\{0, \ldots, m-1\}$.

Let $\mathbb{D}$ be any set among $\mathbb{Q}$, $\mathbb{N}$, $\mathbb{Z}$ and $\mathbb{Z}_m$. For $n \in \mathbb{N}$, $n \geq 1$, we denote by $\mathbb{D}^n$ the set of vectors with n components in $\mathbb{D}$. The ith component of a vector $\mathbf{a}$ is written $\mathbf{a}[i]$. Vector addition, vector multiplication by a scalar and scalar product are defined as usual, i.e.

- given $k \in \mathbb{D}$ and $\mathbf{a}, \mathbf{b} \in \mathbb{D}^n$, $\mathbf{b} = k\mathbf{a}$ is such that $\mathbf{b}[i] = k\mathbf{a}[i]$ for all $i \in \{1, \ldots, n\}$;
- given $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathbb{D}^n$, $\mathbf{c} = \mathbf{a} + \mathbf{b}$ is such that $\mathbf{c}[i] = \mathbf{a}[i] + \mathbf{b}[i]$ for all $i \in \{1, \ldots, n\}$;
- given $k \in \mathbb{D}$ and $\mathbf{a}, \mathbf{b} \in \mathbb{D}^n$, $k = \mathbf{a}.\mathbf{b}$ is such that $k = \sum_{i=1}^{n} \mathbf{a}[i]\mathbf{b}[i]$.

Also, given two sets $A, B \subseteq \mathbb{D}^n$, we define $A + B = \{\mathbf{a} + \mathbf{b} \mid \mathbf{a} \in A \wedge \mathbf{b} \in B\}$. If $A = \{\mathbf{a}\}$, we simply write $\mathbf{a} + B$ for $A + B$. For any vector $\mathbf{a} \in \mathbb{D}^n$, $\| \mathbf{a}^+ \| = \sum_{\mathbf{a}[i] \geq 0} \mathbf{a}[i]$ and $\| \mathbf{a}^- \| = -\sum_{\mathbf{a}[i] < 0} \mathbf{a}[i]$.

For $p, q \in \mathbb{N}$, $p, q \geq 1$, $\mathbb{D}^{p \times q}$ is the set of $p \times q$-matrices with components in $\mathbb{D}$. For a matrix $\mathbf{A} \in \mathbb{D}^{p \times q}$, the row index set of $\mathbf{A}$ is $\{1, \ldots, p\}$ and the column index set is $\{1, \ldots, q\}$, and the entry located in the ith row and jth column is written $\mathbf{A}[i, j]$. The ith row of $\mathbf{A}$ is denoted $\mathbf{A}[i, *]$ and similarly, the jth column is denoted $\mathbf{A}[*, j]$. For $\mathbf{c}_1, \ldots, \mathbf{c}_q \in \mathbb{D}^p$ and $\mathbf{A} \in \mathbb{D}^{p \times q}$, we write $\mathbf{A} = (\mathbf{c}_1 \cdots \mathbf{c}_q)$ if $\mathbf{A}[i, j] = \mathbf{c}_j[i]$ for all $i \in \{1, \ldots, p\}$ and $j \in \{1, \ldots, q\}$. The matrix $\mathbf{I}_n$ will denote in the sequel the identity matrix of dimension n, that is, $\mathbf{I}_n \in \mathbb{Z}^{n \times n}$ and $\mathbf{I}_n[i, j] = \begin{cases} 0 \text{ if } i \neq j \\ 1 \text{ if } i = j \end{cases}$.

We define matrix transposition, multiplication of matrix by a scalar, matrix addition and matrix multiplication as usual, i.e.

- given $\mathbf{A} \in \mathbb{D}^{p \times q}$, $\mathbf{B} \in \mathbb{D}^{q \times p}$, $\mathbf{B} = \mathbf{A}^{\mathtt{t}}$ is such that $\mathbf{B}[j, i] = \mathbf{A}[i, j]$ for all $i \in \{1, \ldots, p\}$, $j \in \{1, \ldots, q\}$;
- given $k \in \mathbb{D}$, $\mathbf{A}, \mathbf{B} \in \mathbb{D}^{p \times q}$, $\mathbf{B} = k\mathbf{A}$ is such that $\mathbf{B}[i, j] = k\mathbf{A}[i, j]$ for all $i \in \{1, \ldots, p\}$, $j \in \{1, \ldots, q\}$;
- given $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathbb{D}^{p \times q}$, $\mathbf{C} = \mathbf{A} + \mathbf{B}$ is such that $\mathbf{C}[i, j] = \mathbf{A}[i, j] + \mathbf{B}[i, j]$ for all $i \in \{1, \ldots, p\}$, $j \in \{1, \ldots, q\}$;

- given $\mathbf{A} \in \mathbb{D}^{p\times q}$, $\mathbf{B} \in \mathbb{D}^{q\times r}$, $\mathbf{C} \in \mathbb{D}^{p\times r}$, $\mathbf{C} = \mathbf{AB}$ is such that $\mathbf{C}[i,j] = \sum_{k=1}^{q} \mathbf{A}[i,k] \cdot \mathbf{B}[k,j]$ for all $i \in \{1,\ldots,p\}$, $j \in \{1,\ldots,r\}$.

A matrix is in *column echelon form* if the following conditions are satisfied.

1. All zero columns are at the right of the matrix.
2. The first nonzero entry of each nonzero column after the first one occurs below the first non-zero entry of the previous column.

An integer matrix $\mathbf{H} \in \mathbb{Z}^{m\times n}$ is in *Hermite form* if the following conditions are satisfied.

1. $\mathbf{H}$ is in column echelon form.
2. For all $i \in \{1,\ldots,n\}$, if the ith column is a nonzero column and if $\mathbf{H}[k_i,i]$ denotes the first non-zero entry of this column, then $0 < \mathbf{H}[k_i,i]$ and $0 \leq \mathbf{H}[k_i,i'] < \mathbf{H}[k_i,i]$ for all $i' \in \{1,\ldots,i-1\}$.

Example 1. *Let* $\mathbf{A}_1 = \begin{bmatrix} 1 & 4 & 8 \\ 3 & 5 & 2 \\ 8 & 8 & 1 \end{bmatrix}$, $\mathbf{A}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 5 & 0 \\ 8 & 8 & 1 \end{bmatrix}$, $\mathbf{A}_3 = \begin{bmatrix} 1 & 0 & 0 \\ 3 & 5 & 0 \\ 8 & 8 & 9 \end{bmatrix}$. *The matrix* $\mathbf{A}_1$ *is neither in column echelon form nor in Hermite form, the matrix* $\mathbf{A}_2$ *is in column echelon form but not in Hermite form, and finally, the matrix* $\mathbf{A}_3$ *is both in column echelon form and in Hermite form.* □

Finally, an integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ is *prime*[1] if the greatest common divisor of the determinants of the largest square matrices obtained from $\mathbf{A}$ by removing some rows or columns is 1.

2.3 Systems of Linear (In)Equations

Given a vector $\mathbf{a} \in \mathbb{D}^n$ and a scalar $b \in \mathbb{D}$, the formula $\mathbf{a}.\mathbf{x} = b$, where $\mathbf{x} \in \mathbb{D}^n$ is an unknown of the equation, is a *linear equation*, also called *affine relation*, and $\mathbf{a}.\mathbf{x} \leq b$ is a *linear inequation*. A system of linear equations is a conjunction of linear equations $\mathbf{a}_1.\mathbf{x} = b_1 \wedge \ldots \wedge \mathbf{a}_m.\mathbf{x} = b_m$ and is denoted $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}[i,j] = \mathbf{a}_i[j]$ and $\mathbf{b}[i] = b_i$, $i \in \{1,\ldots,m\}$. A system of linear inequations is defined similarly and is denoted $\mathbf{Ax} \leq \mathbf{b}$.

[1]The concept of *prime* matrix as defined here appeared in [Smi61].

2.4 Basic Notions of Abstract Algebra

The following definitions are standard definitions [Fra94].

A *group* $\langle G, * \rangle$ is a set G together with a binary operation $*$ such that the following axioms are satisfied.

- The binary operation $*$ is associative.
- There is an element e in G such that $e * x = x * e = x$ for all $x \in G$. This element is an identity element for $*$ on G.
- For each a in G, there is an element a' in G with the property that $a' * a = a * a' = e$. The element a' is an *inverse of* a with respect to $*$.

A group $\langle G, * \rangle$ is *abelian* if $*$ is commutative.

Example 2. *The set $\mathbb{Z}$ together with the addition $+$ is an abelian group, but the set $\mathbb{Z}$ together with the multiplication $\cdot$ is not a group.* □

A *ring* $\langle R, +, \cdot \rangle$ is a set R together with two binary operations $+$ and $\cdot$ of addition and multiplication defined on R such that the following axioms are satisfied.

- $\langle R, + \rangle$ is an abelian group.
- Multiplication is associative.
- For all $a, b, c \in R$, the left distributive law, $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$, and the right distributive law, $(a + b) \cdot c = (a \cdot c) + (b \cdot c)$ hold.

A ring R in which the multiplication is commutative is a commutative ring. A ring R with a multiplicative identity 1 such that $1 \cdot x = x \cdot 1 = x$ for all $x \in R$ is a *ring with unity*. A multiplicative identity in a ring is *unity*. A *multiplicative inverse* of an element a in a ring R with unity 1 is an element $a^{-1} \in R$ such that $a \cdot a^{-1} = a^{-1} \cdot a = 1$.

Let R be a ring with unity. If every nonzero element of R has an inverse, then R is a *division ring*. A *field* is a commutative division ring.

Example 3. *The set $\mathbb{Z}$ together with the addition and multiplication is a commutative ring with unity but it is not a field, whereas the set $\mathbb{Q}$ with the addition and the multiplication is a field.* □

A *vector space* is an abelian group V under addition and a field F, together with an operation of scalar multiplication $\cdot$ of each element of V by each element of F on the left, such that for all $a, b \in F$ and $\mathbf{x}, \mathbf{y} \in V$, the following conditions are satisfied.

- $a \cdot \mathbf{x} \in V$.
- $a \cdot (b \cdot \mathbf{x}) = (a \cdot b) \cdot \mathbf{x}$.
- $(a + b) \cdot \mathbf{x} = (a \cdot \mathbf{x}) + (b \cdot \mathbf{x})$.
- $a \cdot (\mathbf{x} + \mathbf{y}) = (a \cdot \mathbf{x}) + (a \cdot \mathbf{y})$.
- $1 \cdot \mathbf{x} = \mathbf{x}$.

In the sequel, we will say that V is a vector space over F.

A *(left) R-module* is an abelian group M under addition and a ring R, together with an operation of scalar multiplication of each element of M by each element of R on the left such that for all $a, b \in R$ and $\mathbf{x}, \mathbf{y} \in M$, the following conditions are satisfied.

- $a \cdot \mathbf{x} \in M$.
- $a \cdot (b \cdot \mathbf{x}) = (a \cdot b) \cdot \mathbf{x}$.
- $(a + b) \cdot \mathbf{x} = (a \cdot \mathbf{x}) + (b \cdot \mathbf{x})$.
- $a \cdot (\mathbf{x} + \mathbf{y}) = (a \cdot \mathbf{x}) + (a \cdot \mathbf{y})$.

In the sequel, we will say that M is a R-module.

Example 4. *The set $\{k \cdot (1, 1) \mid k \in \mathbb{Z}\}$ with vector addition forms an abelian group, and this group combined with multiplication over $\mathbb{Z}$ forms a $\mathbb{Z}$-module, but not a vector space since $\mathbb{Z}$ is not a field. The set $\{k \cdot (1, 1) \mid k \in \mathbb{Q}\}$ with vector addition forms an abelian group, and this group combined with multiplication over $\mathbb{Q}$ forms a vector space over $\mathbb{Q}$.* □

Note that a R-module is very much like a vector space except that the "scalars" form a ring. Intuitively, the major difference is that in a R-module, one cannot "cancel" a scalar by multiplying by another scalar. For example, one cannot generate $(3, 1) \in \mathbb{Z}^2$ by multiplying $7 \cdot (3, 1) \in \mathbb{Z}^2$ by some integer. Conversely, one obtains $(3, 1) \in \mathbb{Q}^2$ by multiplying $7 \cdot (3, 1) \in \mathbb{Q}^2$ by $\frac{1}{7} \in Q$.

2.5 Quantifier-elimination in Presburger Arithmetic

In this section, we detail an important aspect of the proof in [Pre29, Pre91], i.e. the fact there exists a quantifier-elimination method in Presburger arithmetic. We mainly follow [End01].

First note that the theory of $\langle \mathbb{Z}, 0, 1, +, < \rangle$ does not admit elimination of quantifiers. We can overcome this by adding the symbols $\equiv_2, \equiv_3, \ldots$ with $\equiv_m$ denoting congruence relation modulo m. The structure for this expanded language is then $\langle \mathbb{Z}, 0, 1, +, <, \equiv_{m>0} \rangle$. By definition, for all x, y, we have

$$x \equiv_m y \text{ iff } \exists z \ (x + m \cdot z = y).$$

We deduce that for all sets $S \subseteq \mathbb{Z}^n$, S is definable in the structure $\langle \mathbb{Z}, 0, 1, +, < \rangle$ iff S is definable in $\langle \mathbb{Z}, 0, 1, +, <, (\equiv_m)_{m>0} \rangle$.

In the sequel, we call *Presburger formula* any well-formed formula over the language of the structure $\langle \mathbb{Z}, 0, 1, +, <, (\equiv_m)_{m>0} \rangle$.

Proposition 5. *The theory $\langle \mathbb{Z}, 0, 1, +, <, (\equiv_m)_{m>0} \rangle$ admits the elimination of the quantifiers.*

Proof. We prove that for each formula, there exists an equivalent quantifier-free formula.

Note first that for all formulas φ, $\forall x(\varphi)$ is equivalent to $\neg\exists x \ (\neg\varphi)$, and therefore, one only has to consider the elimination of existential quantification $\exists x$. We detail below a procedure removing the innermost existential quantifier, i.e. given a formula $\exists y \ \varphi(x_1, \ldots, x_n, y)$ where φ is a quantifier-free formula, we generate a quantifier-free formula $\varphi'(x_1, \ldots, x_n)$ equivalent to $\exists y \ \varphi(x_1, \ldots, x_n, y)$. By definition, for all formulas φ_1, φ_2, the formulas $\varphi_1 \Rightarrow \varphi_2$ and $\varphi_1 \Leftrightarrow \varphi_2$ are respectively equivalent to the formulas $\neg\varphi_1 \vee \varphi_2$ and $(\varphi_1 \wedge \varphi_2) \vee (\neg\varphi_1 \wedge \neg\varphi_2)$. Therefore, without loss of generality, we can assume that the only Boolean connectives occurring in φ are $\wedge$, $\vee$ and $\neg$.

For a term t and a natural number n, we denote by $n \cdot t$ and n the terms $\underbrace{t + \ldots + t}_{n}$ and $\underbrace{1 + \ldots + 1}_{n}$ respectively. Any term can be expanded to $a_1 \cdot x_1 + \ldots + a_n \cdot x_n + b$ with $a_1, \ldots, a_n, b \in \mathbb{N}$. Also, we use the abbreviation $t_1 \leq t_2$ to denote $t_1 = t_2 \vee t_1 < t_2$. Since each equality $t_1 = t_2$ can be expressed as $t_1 \leq t_2 \wedge t2 \leq t_1$, we assume in the following that all atomic formulas occurring the formula are either of the form $t_1 \leq t_2$ or of the form $t_1 \equiv_m t_2$. Also, without loss of generality, each variable occurs either on the left-hand side or on the right-hand side but not both.

1. **Eliminate negation**. Thanks to de Morgan's laws, the negations can be pushed inwards, i.e. φ can be transformed into Boolean combination of atomic formulas or negations of atomic formulas. Then, since for all terms t_1, t_2, the formulas $\neg(t_1 \leq t_2)$ and $\neg(t_1 \equiv_m t_2)$ are respectively equivalent to the formulas $t_2 + 1 \leq t_1$ and $\bigvee_{k\in\{1,\dots m_*-1\}}(t_1 \equiv_m t_2 + k)$, one can remove negations. Also, using de Morgan's laws, φ can be transformed into a disjunction of conjunctions of atomic formulas. Note that for all formulas φ_1, φ_2, $\exists y\ (\varphi_1 \vee \varphi_2)$ is equivalent to $(\exists y\ \varphi_1) \vee (\exists y\ \varphi_2)$. So, in the remaining steps of the procedure, it suffices to show how to transform a conjunction of atomic formulas into an equivalent quantifier-free formula. Without loss of generality we can assume that y appears in each conjunct.

   We illustrate the remaining steps with the formula

$$\exists y\ (y \geq 0\ \wedge\ x_1 + x_2 \leq 2 \cdot y + 3\ \wedge\ x_1 + 1 \leq 2 \cdot x_2 + 3 \cdot y\ \wedge\ y \leq x_1 \\ \wedge\ x_1 + x_2 + 1 \equiv_3 2 \cdot y).$$

2. **Uniformize the coefficients of** y. Let $a_* > 0$ be the least common multiple of the coefficient of y. Each equation and inequation can be converted to an equivalent formula in which the coefficient of y is a_* by multiplying the terms by the appropriate factor. For the congruence relations, one has also to multiply the modulos :

$$t_1 \equiv_m t_2 \text{ iff } k \cdot t_1 \equiv_{k\cdot m} k \cdot t_2,$$

   for all terms t_1, t_2 and integer numbers $k, m > 0$.

   In our example, we get

$$\exists y\ (6 \cdot y \geq 0\ \wedge\ 3 \cdot x_1 + 3 \cdot x_2 \leq 6 \cdot y + 9\ \wedge\ 2 \cdot x_1 + 2 \leq 4 \cdot x_2 + 6 \cdot y \\ \wedge\ 6 \cdot y \leq 6 \cdot x_1\ \wedge\ 3 \cdot x_1 + 3 \cdot x_2 + 3 \equiv_9 6 \cdot y).$$

3. **Eliminate the coefficients of** y. Replace $a_* \cdot y$ by y' in each atomic formula and add the congruence relation $y' \equiv_{a_*} 0$ as a new conjunct. By transposing terms to compensate for the absence of substraction, we get a formula of the form

$$\exists y' \left(\bigwedge_{j\in\{1,\dots,p\}} t_j - s_j \leq y'\ \wedge \bigwedge_{j\in\{p+1,\dots,p+q\}} y' \leq t_j - s_j \right. \\ \left. \wedge \bigwedge_{j\in\{p+q+1,\dots,p+q+r\}} y' \equiv_{m_j} t_j - s_j \right),$$

where t_j, s_j, with $1 \leq j \leq p+q+r$, are terms in which y' does not occur.

In our continuing example we get

$$\exists y' \, (0 \leq y' \ \wedge \ 3 \cdot x_1 + 3 \cdot x_2 - 9 \leq y' \ \wedge \ 2 \cdot x_1 - 4 \cdot x_2 + 2 \leq y' \\ \wedge \, y' \leq 6 \cdot x_1 \ \wedge \ y' \equiv_9 3 \cdot x_1 + 3 \cdot x_2 + 3 \ \wedge \ y' \equiv_6 0).$$

We now have a formula that asserts the existence of an integer number y' which is not smaller than certain lower bounds $\alpha_1, \ldots, \alpha_p$ and not larger than certain upper bounds and which satisfies certain congruences. Let m_* be the least common multiple of the moduli $m_{p+q+1}, \ldots, m_{p+q+r}$. By definition, for all $y' \in \mathbb{Z}^n$, $y' + m_* \equiv_{m_j} y'$ for all $j \in \{p+q+1, \ldots, p+q+r\}$. So, as y' increases, the pattern of residues of y' modulo $m_{p+q+1}, \ldots, m_{p+q+r}$ has period m_*. Thus in searching for a solution to the congruences, one only needs to search m_* consecutive integers. Therefore, considering the lower bounds, if there is a solution, then one of the following is a solution :

$$\begin{array}{l} \alpha_1, \alpha_1 + 1, \ldots, \alpha_1 + m_* - 1, \\ \alpha_2, \alpha_2 + 1, \ldots, \alpha_2 + m_* - 1, \\ \ldots \\ \alpha_p, \alpha_p + 1, \ldots, \alpha_p + m_* - 1. \end{array}$$

The formula asserting the existence of a solution for y' can now be replaced by a quantifier-free disjunction that asserts that one of the numbers in the above matrix is an integer solution :

$$\bigvee_{k \in \{0,\ldots,m-1\}} \bigvee_{j' \in \{1,\ldots,p\}} \left(\bigwedge_{j \in \{1,\ldots,j'-1,j'+1,\ldots,p\}} t_j - s_j \leq t_{j'} - s_{j'} + k \right. \\ \wedge \bigwedge_{j \in \{p+1,\ldots,p+q\}} t_{j'} - s_{j'} + k \leq t_j - s_j \\ \left. \wedge \bigwedge_{j \in \{p+q+1,\ldots,p+q+r\}} t_{j'} - s_{j'} + k \equiv_{m_j} t_j - s_j \right).$$

In our continuing example, we get

$$\begin{aligned}\bigvee_{k\in\{0,\ldots,17\}} \big[&(3\cdot x_1+3\cdot x_2-9\le k \ \wedge\ 2\cdot x_1-4\cdot x_2+2\le k \ \wedge\ k\le 6\cdot x_1\\ &\wedge k\equiv_9 3\cdot x_1+3\cdot x_2+3 \ \wedge\ k\equiv_6 0)\\ \vee\ &(0\le 3\cdot x_1+3\cdot x_2-9+k \ \wedge\ 2\cdot x_1-4\cdot x_2+2\le 3\cdot x_1+3\cdot x_2-9+k\\ &\wedge 3\cdot x_1+3\cdot x_2-9+k\le 6\cdot x_1\\ &\wedge 3\cdot x_1+3\cdot x_2-9+k\equiv_9 3\cdot x_1+3\cdot x_2+3 \ \wedge\ 3\cdot x_1+3\cdot x_2-9+k\equiv_6 0)\\ \vee\ &(0\le 2\cdot x_1-4\cdot x_2+2+k \ \wedge\ 3\cdot x_1+3\cdot x_2-9\le 2\cdot x_1-4\cdot x_2+2+k\\ &\wedge 2\cdot x_1-4\cdot x_2+2+k\le 6\cdot x_1\\ &\wedge 2\cdot x_1-4\cdot x_2+2+k\equiv_9 3\cdot x_1+3\cdot x_2+3 \ \wedge\ 2\cdot x_1-4\cdot x_2+2+k\equiv_6 0)\big].\end{aligned}$$

□

2.6 Size and Complexity

We define the size of numbers as follows. The size of an integer number $a \in \mathbb{Z}$, denoted $\mathrm{size}(a)$, is 1 if $a = 0$, and $1 + \lfloor \log_2 |a| \rfloor$ otherwise. The size of a rational a/b, denoted $\mathrm{size}(a/b)$, where $a \in \mathbb{Z}$, $b \in \mathbb{N} \setminus \{0\}$ and $\gcd(a, b) = 1$ is $\mathrm{size}(a) + \mathrm{size}(b)$. The size of a $m \times n$-matrix A, denoted $\mathrm{size}(A)$ is $mn + \sum_{i,j} \mathrm{size}(A[i, j])$.

In order to reason about the complexity of the algorithms presented in this thesis, we assume that direct memory accesses are performed in constant time and that arithmetic operations are performed in unit time.

Chapter 3

Basic Algebra

3.1 Hulls in $\mathbb{Q}$, $\mathbb{Z}$ and $\mathbb{Z}_m$

Let $\mathbb{D}$ be either $\mathbb{Q}$, $\mathbb{Z}$ or $\mathbb{Z}_m$, and let $S \subseteq \mathbb{D}^n$.

A vector $\mathbf{x} \in \mathbb{D}^n$ is a *linear combination over* $\mathbb{D}$ of the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_k \in \mathbb{D}^n$ if $\mathbf{x} = a_1\mathbf{x}_1 + \cdots + a_k\mathbf{x}_k$, for some $a_1, \ldots, a_k \in \mathbb{D}$. By setting additional constraints on the coefficients a_i, one defines the *conic*, *affine* and *convex combinations over* $\mathbb{D}$:

- if $a_1, \ldots, a_k \geq 0$, then $\mathbf{x}$ is a *conic combination over* $\mathbb{D}$,
- if $a_1 + \ldots + a_k = 1$, then $\mathbf{x}$ is an *affine combination over* $\mathbb{D}$, and finally,
- if $a_1, \ldots, a_k \geq 0$ and $a_1 + \ldots + a_k = 1$, then $\mathbf{x}$ is a *convex combination over* $\mathbb{D}$.

For a nonempty subset $S \subseteq \mathbb{D}^n$, the *linear* (resp. *conic*, *affine*, *convex*) *hull of* S over $\mathbb{D}$ is the set of all linear (resp. conic, affine, convex) combinations over $\mathbb{D}$ of finitely many vectors of S, and is denoted by $\mathrm{lin}_{\mathbb{D}}(S)$ (resp. $\mathrm{cone}_{\mathbb{D}}(S)$, $\mathrm{aff}_{\mathbb{D}}(S)$, $\mathrm{conv}_{\mathbb{D}}(S)$).

Proposition 6. *Let $S \subseteq \mathbb{D}^n$. For all $\mathbf{a} \in S$,*

1. $\mathbf{a} + \mathrm{lin}_{\mathbb{D}}(-\mathbf{a} + S) = \mathrm{aff}_{\mathbb{D}}(S)$,
2. $-\mathbf{a} + \mathrm{aff}_{\mathbb{D}}(S) = \mathrm{lin}_{\mathbb{D}}(-\mathbf{a} + S)$.

Proof.

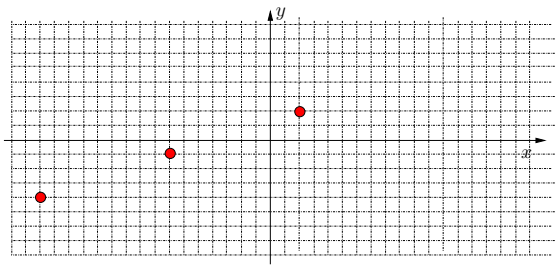


Figure 3.1: $S = \{(-16, -4), (-7, -1), (2, 2)\}$

1. Suppose $\mathbf{x} \in \mathbf{a} + \text{lin}_{\mathbb{D}}(-\mathbf{a} + S)$. By definition, $\mathbf{x} = \mathbf{a} + \sum_{i=1}^{p} k_i(\mathbf{y}_i - \mathbf{a})$, for some $p \in \mathbb{N}$ with $\mathbf{y}_i \in S$ and $k_i \in \mathbb{D}$ for all $i \in \{1, \ldots, p\}$. So, $\mathbf{x} = (1 - \sum_{i=1}^{p} k_i) \cdot \mathbf{a} + \sum_{i=1}^{p} k_i \mathbf{y}_i$, and by definition, $\mathbf{x} \in \text{aff}_{\mathbb{D}}(S)$.

   Conversely, suppose that $\mathbf{x} \in \text{aff}_{\mathbb{D}}(S)$. By definition, $\mathbf{x} = \sum_{i=1}^{p} k_i \mathbf{y}_i$ for some $p \in \mathbb{N}$, with $\mathbf{y}_i \in S$, $k_i \in \mathbb{D}$ for all $i \in \{1, \ldots, p\}$ and $\sum_{i=1}^{p} k_i = 1$. Therefore, $\mathbf{x} = \mathbf{a} + \sum_{i=1}^{p} k_i(\mathbf{y}_i - \mathbf{a})$, and $\mathbf{x} \in \mathbf{a} + \text{lin}_{\mathbb{D}}(-\mathbf{a} + S)$.

2. This is a direct consequence of the above result. Indeed, we have $-\mathbf{a} + \text{aff}_{\mathbb{D}}(S) = -\mathbf{a} + \big(\mathbf{a} + \text{lin}_{\mathbb{D}}(-\mathbf{a} + S)\big) = \text{lin}_{\mathbb{D}}(-\mathbf{a} + S)$. □

Example 7. *Given the set* $S = \{(-16, -4), (-7, -1), (2, 2)\}$*, displayed in Fig.3.1, we have* $\text{cone}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 4y \leq 0 \wedge x - y \leq 0\}$, $\text{cone}_{\mathbb{Z}}(S) = \{a_1 \cdot (-16, -4) + a_2 \cdot (-7, -1) + a_3 \cdot (2, 2) \mid a_1, a_2, a_3 \in \mathbb{N}\}$, $\text{aff}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 3y = -4\}$, $\text{aff}_{\mathbb{Z}}(S) = \{(x, y) \in \mathbb{Z}^2 \mid x - 3y = -4 \wedge x \equiv_9 2\}$, $\text{conv}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 3y = -4 \wedge -16 \leq x \leq 2\}$ *and* $\text{conv}_{\mathbb{Z}}(S) = S = \{(-16, -4), (-7, -1), (2, 2)\}$*. The sets* $\text{cone}_{\mathbb{Q}}(S)$, $\text{aff}_{\mathbb{Q}}(S)$, $\text{conv}_{\mathbb{Q}}(S)$, $\text{aff}_{\mathbb{Z}}(S)$, $\text{conv}_{\mathbb{Z}}(S)$ *and* $\text{cone}_{\mathbb{Z}}(S)$ *are given in Fig.3.2, Fig.3.3, Fig.3.4, Fig.3.5, Fig.3.6 and Fig.3.7 respectively.* □

The vectors $\mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathbb{D}^n$ are *linearly independent over* $\mathbb{D}$ if $\sum_{i=1}^{p} a_i \mathbf{x}_i = \mathbf{0}$ implies that $a_i = 0$, $\forall i \in \{1, \ldots, p\}$. The vectors $\mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathbb{D}^n$ are *affinely independent over* $\mathbb{D}$ if the vectors $\mathbf{x}_1 - \mathbf{x}_j, \ldots, \mathbf{x}_{j-1} - \mathbf{x}_j, \mathbf{x}_{j+1} - \mathbf{x}_j, \ldots, \mathbf{x}_p - \mathbf{x}_j$ are linearly independent over $\mathbb{D}$ for any $j \in \{1, \ldots, p\}$.

If the vectors $\mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathbb{D}^n$ are not linearly (resp. affinely) independent, they are *linearly* (resp. *affinely*) *independent*.

Let $S \subseteq \mathbb{D}^n$. A set G *$\mathbb{D}$-generates* S if $\text{lin}_{\mathbb{D}}(G) = S$. If in addition, the vectors in G are linearly independent over $\mathbb{D}$, then G is a $\mathbb{D}$-basis of S.

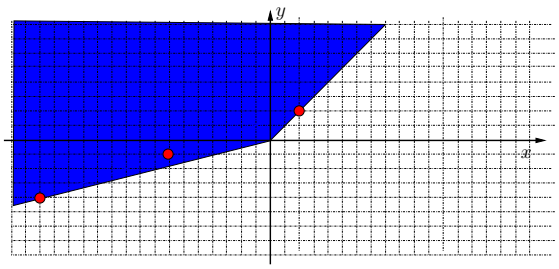


Figure 3.2: $\text{cone}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 4y \leq 0 \wedge x - y \leq 0\}$

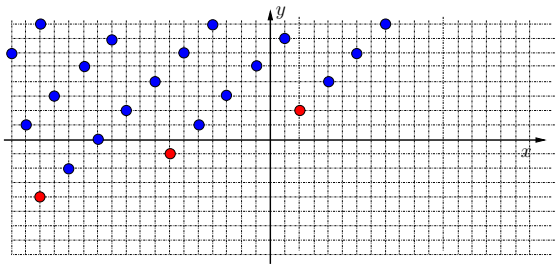


Figure 3.3: $\text{cone}_{\mathbb{Z}}(S) = \{a_1 \cdot (-16, -4) + a_2 \cdot (-7, -1) + a_3 \cdot (2, 2) \mid a_1, a_2, a_3 \in \mathbb{N}\}$

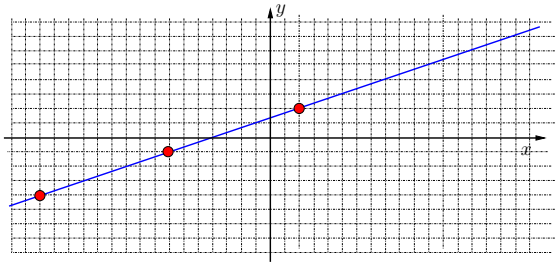


Figure 3.4: $\text{aff}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 3y = -4\}$

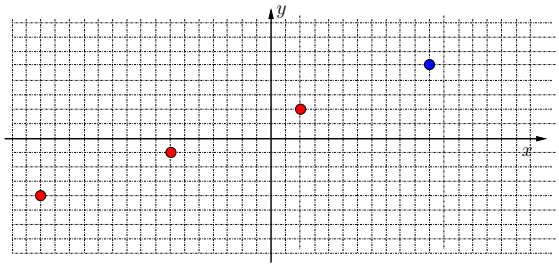


Figure 3.5: $\text{aff}_{\mathbb{Z}}(S) = \{(x, y) \in \mathbb{Z}^2 \mid x - 3y = -4 \wedge x \equiv_9 2\}$

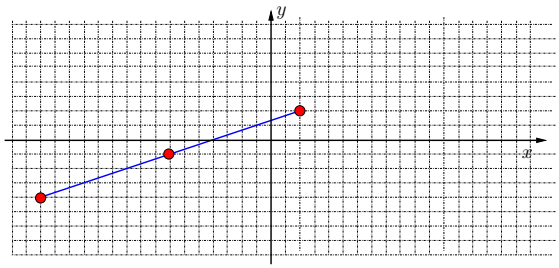


Figure 3.6: $\text{conv}_{\mathbb{Q}}(S) = \{(x, y) \in \mathbb{Q}^2 \mid x - 3y = -4 \wedge -16 \leq x \leq 2\}$

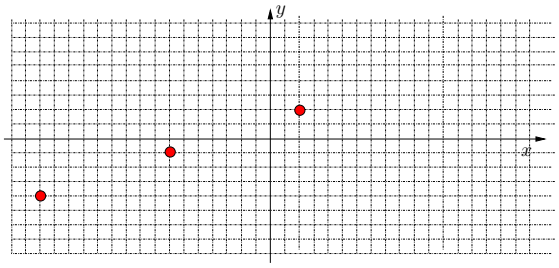


Figure 3.7: $\text{conv}_{\mathbb{Z}}(S) = S = \{(-16, -4), (-7, -1), (2, 2)\}$

Let $S \subseteq \mathbb{D}^n$ and let d be the maximal number of affinely independent vectors in S. The *dimension* of S is $d-1$.

The *rank* of a matrix $\mathbf{A} \in \mathbb{D}^{m \times n}$ is the maximum number of linearly independent columns in $\mathbf{A}$ when each column is considered as a vector.

3.2 Vector Space over $\mathbb{Q}$, Affine Space over $\mathbb{Q}$

Proposition 8. *A set $V \subseteq \mathbb{Q}^n$ is a vector space over $\mathbb{Q}$ iff $V \neq \emptyset$ and $\mathrm{lin}_{\mathbb{Q}}(V) = V$.*

Proof. Direct consequence of the definitions. □

By analogy, we define an *affine space* over $\mathbb{Q}$ as a nonempty set $A \subseteq \mathbb{Q}^n$ such that $\mathrm{aff}_{\mathbb{Q}}(A) = A$.

Proposition 9. *Let $S \subseteq \mathbb{Q}^n$. There exists a unique vector space V over $\mathbb{Q}$ such that $\mathrm{aff}_{\mathbb{Q}}(S) = \mathbf{a} + V$ for some $\mathbf{a} \in \mathbb{Q}^n$.*

Proof. Let $\mathbf{a} \in S$ and $V = \mathrm{lin}_{\mathbb{Q}}(-\mathbf{a} + S)$. Thanks to Proposition 6, $\mathrm{aff}_{\mathbb{Q}}(S) = \mathbf{a} + V$.

Suppose that $\mathrm{aff}_{\mathbb{Q}}(S) = \mathbf{a}' + V'$ for some $\mathbf{a}' \in \mathbb{Q}^n$ and a vector space $V' \subseteq \mathbb{Q}^n$. By construction, $\mathbf{a} + V = \mathbf{a}' + V'$. By definition, $\mathbf{0} \in V \cap V'$, and therefore, $\mathbf{a} = \mathbf{a}' + \mathbf{x}'$ for some $\mathbf{x}' \in V'$ and $\mathbf{a}' = \mathbf{a} + \mathbf{x}$ for some $\mathbf{x} \in V$. So, we have

$$V = -\mathbf{a} + (\mathbf{a} + V) = -\mathbf{a} + (\mathbf{a}' + V') = -\mathbf{x}' + V'.$$

Since V' is a vector space and $\mathbf{x}' \in V'$, we have $-\mathbf{x}' + V' = V'$ and we conclude that $V = V'$. □

Proposition 10. *Any vector space $S \subseteq \mathbb{Q}^n$ has a $\mathbb{Q}$-basis, and all $\mathbb{Q}$-bases of S have the same number of elements $d \leq n$. If the set $\{\mathbf{x}_1, \ldots, \mathbf{x}_k\} \subseteq S$ is a set of linearly independent vectors, then it can be* enlarged *to form a $\mathbb{Q}$-basis of S, that is, there exist $\mathbf{y}_1, \ldots, \mathbf{y}_t \in S$ such that $\{\mathbf{x}_1, \ldots, \mathbf{x}_k, \mathbf{y}_1, \ldots, \mathbf{y}_t\}$ is a $\mathbb{Q}$-basis of S.*

Proof. See [Fra94]. □

Example 11. *The set $S = \{(x, y) \in \mathbb{Q}^2 \mid 2x - y = 1\}$ is a $\mathbb{Q}$-affine space. The vector space V associated to S is $V = \{(x, y) \in \mathbb{Q}^2 \mid 2x - y = 0\}$, i.e. $V = \{(a, 2a) \mid a \in \mathbb{Q}\}$. Note that $S = (a, 2a - 1) + V$ for all $a \in \mathbb{Q}$. Finally, $\{(1, 2)\}$ is a $\mathbb{Q}$-basis of V.* □

Proposition 12. *Any sequence of vector spaces $V_1, V_2, \ldots \subseteq \mathbb{Q}^n$ such that $V_1 \subset V_2 \subset \ldots$ is finite and bounded by $n+1$.*

Proof. By definition, $\dim(V_i) < \dim(V_{i+1})$ and $0 \leq \dim(V_i) \leq n$ for all i, and therefore, there are at most $n+1$ vector spaces in the sequence. □

3.3 $\mathbb{Z}$- and $\mathbb{Z}_m$-Modules, $\mathbb{Z}$- and $\mathbb{Z}_m$-Affine Modules

Let $\mathbb{D}$ be either $\mathbb{Z}$ or $\mathbb{Z}_m$.

Proposition 13. *A set $M \subseteq \mathbb{Q}^n$ is a $\mathbb{D}$-module iff $M \neq \emptyset$ and $\mathrm{lin}_{\mathbb{D}}(M) = M$.*

Proof. Direct consequence of the definitions. □

By analogy, we define an *affine $\mathbb{D}$-module* as a nonempty set $A \subseteq \mathbb{Q}^n$ such that $\mathrm{aff}_{\mathbb{D}}(A) = A$.

Proposition 14. *Let $S \subseteq \mathbb{Q}^n$. There exists a unique $\mathbb{D}$-module M such that $\mathrm{aff}_{\mathbb{D}}(S) = \mathbf{a} + M$ for some $\mathbf{a} \in \mathbb{D}^n$.*

Proof. The proof is similar to the proof of Proposition 9. □

Proposition 15. *Any $\mathbb{D}$-module $S \subseteq \mathbb{D}^n$ has a $\mathbb{D}$-basis, and all $\mathbb{D}$-basis of S have the same number of elements $d \leq n$.*

Proof. See [Jac89]. □

Proposition 16. *For all $k \in \mathbb{N}$, there exist a sequence of $\mathbb{Z}$-modules $M_1, \ldots M_{k+1} \subseteq \mathbb{Z}^n$ such that $M_1 \subset M_2 \subset \ldots \subset M_{k+1}$.*

Proof. Indeed, for any vector $\mathbf{g} \in \mathbb{Z}^n \setminus \{\mathbf{0}\}$, the sequence $M_1, \ldots, M_{k+1} \subseteq \mathbb{Z}^n$ where

$$M_i = \mathrm{lin}_{\mathbb{Z}}(2^{k+1-i} \cdot \mathbf{g}),$$

is such that $M_i \subset M_{i+1}$ for all $i \in \{1, \ldots, k\}$. □

Proposition 17. *Any sequence of $\mathbb{Z}$-modules $M_1, M_2, \ldots \subseteq \mathbb{Z}^n$ such that $M_1 \subset M_2 \subset \ldots$ is finite.*

Proof. See [Gra91]. □

Propositions 12 and 16 emphasize a major difference between $\mathbb{Z}$-modules and vector spaces over $\mathbb{Q}$. Another difference that will be relevant in the sequel is that any set of linearly independent vectors of a vector space V over $\mathbb{Q}$ can enlarged to form a $\mathbb{Q}$-basis of V, whereas it is in general not possible to enlarge a set of linearly independent vectors of a $\mathbb{Z}$-module M to form a $\mathbb{Z}$-basis of M.

Example 18. *The set $S = \{(x, y) \in \mathbb{Z}^2 \mid 2x - y = 1\}$ is an affine $\mathbb{Z}$-module. The $\mathbb{Z}$-module M associated to S is $M = \{(x, y) \in \mathbb{Z}^2 \mid 2x - y = 0\}$, i.e. $M = \{(a, 2a) \mid a \in \mathbb{Z}\}$. Note that $S = (a, 2a - 1) + M$ for all $a \in \mathbb{Z}$. Finally, $\{(1, 2)\}$ is a $\mathbb{Z}$-basis of M. Note that the set of (trivially) linearly independent vector $\{(2, 4)\} \subseteq M$ can not be enlarged to form a basis of M.* □

3.4 Polyhedra

In this section, the definition domain is $\mathbb{Q}$, i.e. all scalars and matrix components are rational numbers and the domain of the variables is $\mathbb{Q}$.

An inequation $\mathbf{a}.\mathbf{x} \leq b$ from $\mathbf{Ax} \leq \mathbf{b}$ is called an *implicit equation in* $\mathbf{Ax} \leq \mathbf{b}$ if $\mathbf{Ax} \leq \mathbf{b}$ implies $\mathbf{a}.\mathbf{x} = b$.

Example 19. *The implicit equations in the system $x+2y-z \leq -1 \wedge -x-2y+z \leq 1 \wedge -x \leq 0$ are the inequations $x + 2y - z \leq 1$ and $-x - 2y + z \leq -1$.* □

We use the following notations (taken from [Sch86]) :

- $\mathbf{A}^{=}\mathbf{x} \leq \mathbf{b}^{=}$ is the (possibly empty) subsystem of $\mathbf{Ax} \leq \mathbf{b}$ formed by the implicit equations,
- $\mathbf{A}^{+}\mathbf{x} \leq \mathbf{b}^{+}$ is the (possibly empty) subsystem of $\mathbf{Ax} \leq \mathbf{b}$ formed by the inequations which are not implicit equations.

If all solutions of a proper subsystem $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ satisfy $\mathbf{Ax} \leq \mathbf{b}$, then the inequations of $\mathbf{Ax} \leq \mathbf{b}$ which do not appear in $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ are *redundant*. If a system $\mathbf{Ax} \leq \mathbf{b}$ has no redundant inequation, then it is called *irredundant*. It is always possible to generate an irredundant system $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ from a system $\mathbf{Ax} \leq \mathbf{b}$ by removing successively one redundant inequation until there is no redundant inequation left.

The set $C \subseteq \mathbb{Q}^n$ is a *cone* if $C = \mathrm{cone}_{\mathbb{Q}}(C)$. A cone C is *polyhedral* if $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Ax} \leq \mathbf{0}\}$ for some rational matrix $\mathbf{A}$. The cone C is *generated* by the set G if $C = \mathrm{cone}_{\mathbb{Q}}(G)$. If there exists a finite set G such that $C = \mathrm{cone}_{\mathbb{Q}}(G)$, then C is *finitely generated*.

Theorem 20. *A convex cone is polyhedral iff it is finitely generated.* □

Proof. See [Sch86]. □

A set P of vectors in $\mathbb{Q}^n$ is called a (convex) *polyhedron* if $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ for some rational matrix $\mathbf{A}$ and some rational vector $\mathbf{b}$.

A set of vectors is a (convex) *polytope* if it is the convex hull of finitely many vectors.

Theorem 21. *A set P of vectors is a polyhedron iff $P = Q + C$ for some polytope Q and cone C.*

Proof. See [Sch86]. □

In the remaining of this section, P denotes the polyhedron $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$, with $\mathbf{A} \in \mathbb{Q}^{m \times n}$ and $\mathbf{b} \in \mathbb{Q}^m$.

The *characteristic cone* of P, denoted by $\text{char-cone}(P)$, is the set

$$\begin{aligned} \text{char-cone}(P) &= \{\mathbf{y} \in \mathbb{Q}^n \mid (\forall \mathbf{x} \in P)(\mathbf{x} + \mathbf{y} \in P)\} \\ &= \{\mathbf{y} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{y} \leq 0\}. \end{aligned}$$

The *lineality space* of P, denoted $\text{lin-space}(P)$, is the vector space

$$\begin{aligned} \text{lin-space}(P) &= \{\mathbf{y} \mid \mathbf{y} \in \text{char-cone}(P) \wedge -\mathbf{y} \in \text{char-cone}(P)\} \\ &= \{\mathbf{y} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{y} = 0\}. \end{aligned}$$

If the dimension of the lineality space is zero, P is said to be *pointed*. Note that if P is pointed, then for any polyhedron P' with $P' \subseteq P$, P' is also pointed.

Theorem 22. *If $P = Q + C$ for some polytope Q and cone C, then $C = \text{char-cone}(P)$.*

Proof. See [Sch86]. □

Example 23. *The polyhedron* $P = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} -4 \\ 4 \\ -9 \end{bmatrix}\}$ *is displayed in Figure 3.8. The characteristic cone of* P *is* $\text{char-cone}(P) = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\}$ *and is displayed in Figure 3.9.* □

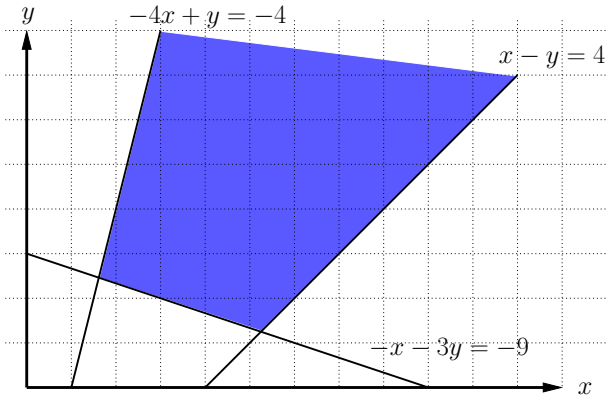


Figure 3.8: The polyhedron $P = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} -4 \\ 4 \\ -9 \end{bmatrix}\}$

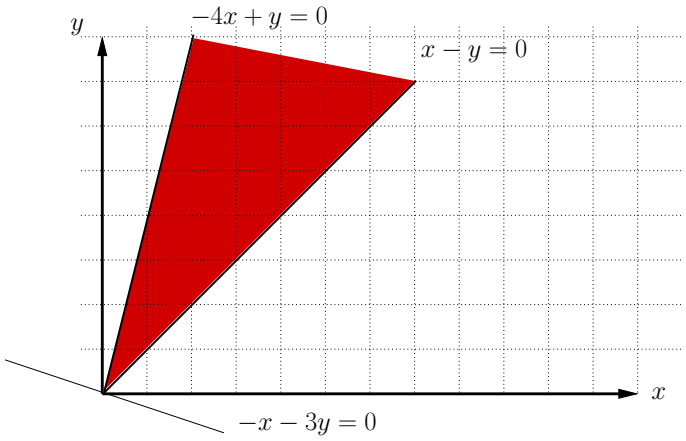


Figure 3.9: The cone $\text{char-cone}(P) = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\}$

Theorem 24. *If P is a convex polyhedron, then $\mathrm{conv}_{\mathbb{Q}}(P \cap \mathbb{Z}^n)$ is also a polyhedron, and $\text{char-cone}(P) = \text{char-cone}\,(\mathrm{conv}_{\mathbb{Q}}(P \cap \mathbb{Z}^n))$.*

Proof. See [Sch86]. □

If $\mathbf{c}$ is a non-zero vector and $b = \max\{\mathbf{c}.\mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ is well-defined, then the set $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}.\mathbf{x} = b\}$ is a *supporting hyperplane* of P. A subset F of P is called a *face* of P if either $F = P$, or F is the intersection of P with a supporting hyperplane of P. A *facet* of P is a maximal (w.r.t. inclusion) face distinct from P.

Theorem 25. *F is a face of P iff $F = \{\mathbf{x} \in P \mid \mathbf{A}'\mathbf{x} = \mathbf{b}'\}$ for some subsystem $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$.*

Proof. See [Sch86]. □

Theorem 26. *If no inequation in $\mathbf{A}^+\mathbf{x} \leq \mathbf{b}^+$ is redundant in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, then there exists a one-to-one correspondence between facets of P and the inequations in $\mathbf{A}^+\mathbf{x} \leq \mathbf{b}^+$, given by*

$$F = \{\mathbf{x} \in P \mid \mathbf{a}.\mathbf{x} = b\}$$

for any facet F of P and any inequation $\mathbf{a}.\mathbf{x} \leq b$ from $\mathbf{A}^+\mathbf{x} \leq \mathbf{b}^+$.

Proof. See [Sch86]. □

Theorem 27. *The dimension of a face F of P is $\dim(P) - 1$ iff F is a facet of P.*

Proof. See [Sch86]. □

In the following lemmas, we assume that C is a cone with $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$ and no inequation in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ is redundant in $\mathbf{C}\mathbf{x} \leq \mathbf{0}$.

Lemma 28. $\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^=\mathbf{x} = 0\}$.

Proof. Since for all $\mathbf{x} \in C$, $\mathbf{C}^=\mathbf{x} = \mathbf{0}$, by definition of the linear hull, $\mathrm{lin}_{\mathbb{Q}}(C) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^=\mathbf{x} = \mathbf{0}\}$.

Suppose now that $\mathbf{y} \in \mathbb{Q}^n$ with $\mathbf{C}^=\mathbf{y} = \mathbf{0}$. If there is no inequation in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$, then $\mathbf{y} \in C$ and $\mathbf{y} \in \mathrm{lin}_{\mathbb{Q}}(C)$. Suppose therefore that there are t inequations in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$, denoted by $\mathbf{c}_1.\mathbf{x} \leq 0, \ldots, \mathbf{c}_t.\mathbf{x} \leq 0$. By definition, for each $i \in \{1, \ldots, t\}$, there exists $\mathbf{y}_i \in C$ such that $\mathbf{c}_i.\mathbf{y}_i = a_i < 0$. Let $\mathbf{c}_i.\mathbf{y} = b_i$ and let

$\mathbf{z} = \sum_{i\in\{1,\ldots,t\}} \left|\frac{b_i}{a_i}\right| \mathbf{y}_i + \mathbf{y}$. By construction, $\mathbf{z} \in C$. Indeed, $\mathbf{C}^{=}\mathbf{z} = \mathbf{0}$ and for all $\mathbf{c}_i.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$, we have

$$\begin{aligned} \mathbf{c}_i.\mathbf{z} &= \mathbf{c}_i.\Big(\sum_{j\in\{1,\ldots,t\}} \left|\frac{b_j}{a_j}\right| \mathbf{y}_j + \mathbf{y}\Big) \\ &\leq \mathbf{c}_i.\Big(\left|\frac{b_j}{a_j}\right| \mathbf{y}_j + \mathbf{y}\Big) \\ &\leq 0 \end{aligned}$$

We conclude that $\mathbf{y} \in \mathrm{lin}_{\mathbb{Q}}(C)$ since $\mathbf{y}$ is a linear combination of elements in $\mathrm{lin}_{\mathbb{Q}}(C)$. □

Lemma 29. *Let F be a facet of C, with $F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\}$ for some inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$.*

$$\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}.$$

Proof. Since for all $\mathbf{x} \in F$, $\mathbf{C}^{=}\mathbf{x} = \mathbf{0}$ and $\mathbf{c}.\mathbf{x} = 0$, by definition of the linear hull, we have

$$\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}. \tag{3.1}$$

Since $\mathbf{c}.\mathbf{x} \leq 0$ is not an implicit equation in $\mathbf{C}\mathbf{x} \leq \mathbf{0}$, there exists $y \in \mathbb{Q}^n$ such that $\mathbf{C}^{=}y = \mathbf{0}$ but $\mathbf{c}.\mathbf{y} \neq 0$, and therefore, $\dim(\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}) = \dim(\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0}\}) - 1$. According to Lemma 28, $\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0}\}$, and therefore, $\dim(\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}) = \dim(C) - 1$. From Theorem 27, $\dim(F) = \dim(C) - 1$, and we deduce that

$$\dim(\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}) = \dim(F) \tag{3.2}$$

From (3.1) and (3.2), and by definition of a linear hull, we conclude that

$$\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0\}. \tag{3.3}$$

□

Lemma 30. *For all faces F_1, F_2 of a cone C, we have*

$$\mathrm{lin}_{\mathbb{Q}}(F_1) = \mathrm{lin}_{\mathbb{Q}}(F_2) \Leftrightarrow F_1 = F_2.$$

Proof. Let $\mathbf{c}_1, \ldots, \mathbf{c}_m \in \mathbb{Q}^n$ such that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge_{i\in\{1,\ldots,m\}} \mathbf{c}_i.\mathbf{x} \leq 0\}$. Thanks to Theorem 25, there exists a set $I_1 \subseteq \{1, \ldots, m\}$ such that

$$F_1 = \{\mathbf{x} \in C \mid \bigwedge_{i\in I_1} \mathbf{c}_i.\mathbf{x} = 0\}.$$

Without loss of generality, we can assume that for all $i \in \{1, \ldots, m\} \setminus I_1$, we have

$$F_1 \not\subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_i.\mathbf{x} = 0\}.$$

Similarly, there exists a set $I_2 \subseteq \{1, \ldots, m\}$ such that

$$F_2 = \{\mathbf{x} \in C \mid \bigwedge_{i\in I_2} \mathbf{c}_i.\mathbf{x} = 0\},$$

and for all $i \in \{1, \ldots, m\} \setminus I_1$, we have

$$F_2 \not\subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_i.\mathbf{x} = 0\}.$$

By definition, $F_1 = F_2$ iff $I_1 = I_2$. Also, F_1 and F_2 are cones and thanks to Lemma 28, we have

$$\begin{aligned} \mathrm{lin}_{\mathbb{Q}}(F_1) &= \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge i \in I_1 \mathbf{c}_i.\mathbf{x} = 0\} \\ \mathrm{lin}_{\mathbb{Q}}(F_2) &= \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge_{i\in I_2} \mathbf{c}_i.\mathbf{x} = 0\}. \end{aligned}$$

We prove that $\mathrm{lin}_{\mathbb{Q}}(F_1) = \mathrm{lin}_{\mathbb{Q}}(F_2)$ iff $I_1 = I_2$.

- Clearly, if $I_1 = I_2$, then $\mathrm{lin}_{\mathbb{Q}}(F_1) = \mathrm{lin}_{\mathbb{Q}}(F_2)$.
- Assume now that $I_1 \neq I_2$. Without loss of generality, there exists $j \in I_2 \setminus I_1$. By hypothesis, $F_2 \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_j.\mathbf{x} = 0\}$ and thus

  $$\mathrm{lin}_{\mathbb{Q}}(F_2) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_j.\mathbf{x} = 0\}.$$

  Conversely, $F_1 \not\subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_j.\mathbf{x} = 0\}$ and so, we have

  $$\mathrm{lin}_{\mathbb{Q}}(F_1) \not\subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{c}_j.\mathbf{x} = 0\}.$$

  We deduce that $\mathrm{lin}_{\mathbb{Q}}(F_1) \neq \mathrm{lin}_{\mathbb{Q}}(F_2)$.

So, we conclude that $F_1 = F_2$ iff $I_1 = I_2$ iff $\mathrm{lin}_{\mathbb{Q}}(F_1) = \mathrm{lin}_{\mathbb{Q}}(F_2)$. □

3.5 Integer Solutions of Systems of Linear Equations

Let $\mathbf{A} \in \mathbb{Z}^{m \times n}$ with $\mathrm{rank}(\mathbf{A}) = r < n$, and let $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}) \in \mathbb{Z}^{n \times (n-r)}$. The set $\{\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}\}$ is a *complete set of solutions* of $\mathbf{Ax} = \mathbf{0}$ if $\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}$ are linearly independent and if $\mathbf{Ax}_i = \mathbf{0}$ for all $\mathbf{x}_i \in \{\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}\}$. If in addition, the matrix X is prime then $\{\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}\}$ is a *fundamental set* of solutions. If $\mathrm{rank}(\mathbf{A}) = n$, then $\mathbf{0}$ is the only solution to the linear system $\mathbf{Ax} = \mathbf{0}$. In this case, the fundamental set of solutions is the empty set.

Theorem 31. *For any matrix $\mathbf{A} \in \mathbb{Z}^{m \times n}$, there exists a fundamental set of solutions $\{\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}\}$, with $r = \mathrm{rank}(\mathbf{A})$, for the system $\mathbf{Ax} = \mathbf{0}$.*

Proof. See [Smi61]. □

Theorem 32. *If $\mathbf{A} \in \mathbb{Z}^{m \times n}$ and $\mathbf{b} \in \mathbb{Z}^m$ are such that the linear system $\mathbf{Ax} = \mathbf{b}$ has at least one integer solution $\mathbf{y} \in \mathbb{Z}^n$, then the set of integer solutions for the linear system is :*

$$\{\mathbf{y} + \sum_{i=1}^{n-r} a_i \mathbf{x}_i \mid a_i \in \mathbb{Z}\}$$

where $r = \mathrm{rank}(\mathbf{A})$ and the set $\{\mathbf{x}_1, \ldots, \mathbf{x}_{n-r}\}$ is a fundamental set of solutions of the linear system $\mathbf{Ax} = \mathbf{0}$.

Proof. See [Smi61]. □

3.6 Hilbert Basis and Integer Elements of Polyhedra

In this section, we describe the concepts of Hilbert basis and of extended Hilbert basis. The concept of (extended) Hilbert basis has been of interest in integer linear programming problems [Sch86], in important unification problems [Kir89] and even in the context of verification [BW01]. Many algorithms have been proposed for generating the (extended) Hilbert basis given a system of linear (in)equations, including [CF89, Pot91, Dom91, AC95, AC97].

Intuitively, an Hilbert basis is a finite set of integer vectors such that the positive integer combinations of those vectors correspond to the integer elements of the cone generated by the set of vectors. Formally, we have the following definition.

Definition 33. *A finite set of integer vectors* $\{\mathbf{y}_1, \ldots, \mathbf{y}_s\}$ *is an* Hilbert basis *if* $\mathrm{cone}_{\mathbb{Q}}(\mathbf{y}_1, \ldots, \mathbf{y}_s) \cap \mathbb{Z}^n = \mathrm{cone}_{\mathbb{Z}}(\mathbf{y}_1, \ldots, \mathbf{y}_s)$.

We have the following theorem regarding Hilbert basis of polyhedral cone.

Theorem 34. *For any polyhedral cone* C*, there exists an Hilbert basis* $\mathbf{y}_1, \ldots, \mathbf{y}_s$ *such that* $C \cap \mathbb{Z}^n = \mathrm{cone}_{\mathbb{Z}}(\mathbf{y}_1, \ldots, \mathbf{y}_s)$.

If C *is pointed, then there is a unique minimal (w.r.t. inclusion) Hilbert basis.*

Proof. See [Hil90, vdC31, GP79, Sch86]. □

Example 35. *The (minimal) Hilbert basis of the set* $C \cap \mathbb{Z}^2$ *with*

$$C = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}\}$$

is $\{(1,1), (1,2), (1,3), (1,4)\}$. □

So, Hilbert basis are the generators of the set of integer solutions of systems of homogeneous inequations, and we can therefore relate this concept to that of generators of the integer solutions of a system of homogeneous equations, i.e. a fundamental set of solutions of the system. An important difference between those concepts is that there exists no bounds on the number of elements in the smallest Hilbert basis whereas the number of elements in the fundamental solutions of a system of linear equations is bounded by the number of variables.

The concept of Hilbert basis can be extended as follows.

Definition 36. *An* extended Hilbert basis *is a pair of finite sets* $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_t\}$ *and* $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_s\}$ *such that* Y *is an Hilbert basis. The elements in* X *are the* constants *and the elements in* Y *are the* periods *of the basis. The set* S generated *from* (X, Y) *is the set* $X + \mathrm{cone}_{\mathbb{Z}}(Y)$.

The extended Hilbert basis is minimal *if* Y *is a minimal Hilbert basis, and if for all* $\mathbf{x} \in X$*,* $\mathbf{x}$ *can not be decomposed into* $\mathbf{x}' \in X \setminus \{\mathbf{x}\}$*, and* $\mathbf{y} \in \mathrm{cone}_{\mathbb{Z}}(Y)$ *such that* $\mathbf{x} = \mathbf{x}' + \mathbf{y}$.

Remark 37. *The concept of extended Hilbert basis is very similar to that of* basis *presented in [AC95, RV02]. A major difference is that an extended Hilbert basis is not defined with respect to a polyhedron, i.e. we do not impose that for any extended Hilbert basis* (X, Y)*,* $X + \mathrm{cone}_{\mathbb{Z}}(Y) = P \cap \mathbb{Z}^n$ *for some polyhedron* P. □

We first present two properties of the extended Hilbert basis, and then we show the relationship between extended Hilbert basis basis and polyhedra.

Lemma 38. *Let (X, Y) and (X, Y') be two extended Hilbert bases.*

If $X + \text{cone}_{\mathbb{Z}}(Y) = X + \text{cone}_{\mathbb{Z}}(Y')$, then $\text{cone}_{\mathbb{Z}}(Y) = \text{cone}_{\mathbb{Z}}(Y')$.

Proof. Without loss of generality, it suffices to prove that for all $\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y)$, $\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y')$.

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_t\}$ and let $\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y)$. By hypothesis, for each $\mathbf{x} \in X$, $\mathbf{x}+\mathbf{y} \in X+\text{cone}_{\mathbb{Z}}(Y')$. We construct recursively the sequences $\mathbf{x}_{k_1}, \mathbf{x}_{k_2}, \ldots \in X$ and $\mathbf{y}_{k_2}, \mathbf{y}_{k_3}, \ldots \in \text{cone}_{\mathbb{Z}}(Y')$ as follows. We choose $\mathbf{x}_{k_1} = \mathbf{x}_1$, and for all $i \geq 1$, $\mathbf{x}_{k_{i+1}}$ and $\mathbf{y}_{k_{i+1}}$ are any vector in X and $\text{cone}_{\mathbb{Z}}(Y')$ respectively such that

$$\mathbf{x}_{k_i} + \mathbf{y} = \mathbf{x}_{k_{i+1}} + \mathbf{y}_{k_{i+1}}.$$

Since X is finite, there is a positive integer m is such that $\mathbf{x}_{k_{m+1}} \in \{\mathbf{x}_{k_1}, \ldots, \mathbf{x}_{k_m}\}$, and for all $1 \leq i \neq j \leq m$, $\mathbf{x}_{k_i} \neq \mathbf{x}_{k_j}$. Let s with $1 \leq s \leq m$ such that $\mathbf{x}_{k_{m+1}} = \mathbf{x}_{k_s}$. By construction, we have

$$\begin{aligned} \sum_{i=s}^{m} \mathbf{x}_{k_i} + (m-s+1)\cdot\mathbf{y} &= \sum_{i=s+1}^{m+1} \mathbf{x}_{k_i} + \sum_{i=s+1}^{m+1} \mathbf{y}_{k_i} \\ &= \sum_{i=s}^{m} \mathbf{x}_{k_i} + \sum_{i=s+1}^{m+1} \mathbf{y}_{k_i}. \end{aligned}$$

So, $(m-s+1)\cdot\mathbf{y} = \sum_{i=s+1}^{m+1} \mathbf{y}_{k_i}$ and $(m-s+1)\cdot\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y')$. By definition, Y' is an Hilbert basis and $\text{cone}_{\mathbb{Z}}(Y') = \text{cone}_{\mathbb{Q}}(Y') \cap \mathbb{Z}^n$. Since $(m-s+1)\cdot\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y')$, by definition, $\mathbf{y} \in \text{cone}_{\mathbb{Q}}(Y')$ and $\mathbf{y} \in \mathbb{Z}^n$. We conclude that $\mathbf{y} \in \text{cone}_{\mathbb{Z}}(Y')$. □

Remark 39. *In general, given a finite set $X \subseteq \mathbb{Z}^n$ and two sets $Y, Y' \subseteq \mathbb{Z}^n$, $X + Y = X + Y'$ does not imply that $Y = Y'$. For example, if $X = \{0, 1\}$, $Y = \{1, 3\}$ and $Y' = \{1, 2, 3\}$, we have $X + Y = X + Y'$ although $Y \neq Y'$.* □

Lemma 40. *Let (X, Y) be an extended Hilbert basis.*

If the cone $\text{cone}_{\mathbb{Q}}(Y)$ is pointed, then there exists a unique minimal extended Hilbert basis $(X_{\min}, Y_{\min})$, with $X_{\min} \subseteq X$ such that

$$X + \text{cone}_{\mathbb{Z}}(Y) = X_{\min} + \text{cone}_{\mathbb{Z}}(Y_{\min}).$$

Proof. Since $\mathrm{cone}_{\mathbb{Q}}(Y)$ is pointed, then there exists a unique minimal Hilbert basis $Y_{\min}$ such that $\mathrm{cone}_{\mathbb{Z}}(Y) = \mathrm{cone}_{\mathbb{Z}}(Y_{\min})$.

Let $X_{\min}$ be computed as follows. Initially, $X_{\min} = X$ and while there exists $\mathbf{x} \in X_{\min}$ such that $\mathbf{x} = \mathbf{x}' + \mathbf{y}$ with $\mathbf{y} \in C \cap \mathbb{Z}^n$ and $\mathbf{x}' \in X_{\min} \setminus \{\mathbf{x}\}$, remove $\mathbf{x}$ from $X_{\min}$.

By definition, $(X_{\min}, Y_{\min})$ is minimal and we have

$$X_{\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = X + \mathrm{cone}_{\mathbb{Z}}(Y).$$

Suppose that there exists another minimal basis (X', Y') such that

$$X' + \mathrm{cone}_{\mathbb{Z}}(Y') = X + \mathrm{cone}_{\mathbb{Z}}(Y).$$

We prove first that $Y_{\min} = Y'$ and then that $X_{\min} = X'$.

1. By definition, we have

$$\begin{aligned} X_{\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) &= X' + \mathrm{cone}_{\mathbb{Z}}(Y') \\ &= X' + \mathrm{cone}_{\mathbb{Z}}(Y') + \mathrm{cone}_{\mathbb{Z}}(Y') \\ &= X_{\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) + \mathrm{cone}_{\mathbb{Z}}(Y') \\ &= X_{\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min} \cup Y')\,. \end{aligned}$$

   From Lemma 38, $\mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = \mathrm{cone}_{\mathbb{Z}}(Y_{\min} \cup Y')$. Similarly, we show that $\mathrm{cone}_{\mathbb{Z}}(Y') = \mathrm{cone}_{\mathbb{Z}}(Y_{\min} \cup Y')$, and we deduce that $\mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = \mathrm{cone}_{\mathbb{Z}}(Y')$. Since $Y_{\min}$ and Y' are both minimal Hilbert basis, $Y_{\min} = Y'$.

2. Let $\mathbf{x} \in X_{\min}$. By hypothesis $\mathbf{x} \in X' + \mathrm{cone}_{\mathbb{Z}}(Y')$ and from above, we have $Y_{\min} = Y'$. So, $\mathbf{x} = \mathbf{x}' + \mathbf{y}$ for some $\mathbf{x}' \in X'$ and $\mathbf{y} \in \mathrm{cone}_{\mathbb{Z}}(Y_{\min})$. Similarly, $\mathbf{x}' \in X_{\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min})$, and therefore, $\mathbf{x}' = \mathbf{x}_2 + \mathbf{y}_2$ for some $\mathbf{x}_2 \in X_{\min}$ and $\mathbf{y}_2 \in \mathrm{cone}_{\mathbb{Z}}(Y_{\min})$. So, $\mathbf{x} = \mathbf{x}_2 + \mathbf{y} + \mathbf{y}_2$ with $\mathbf{x}_2 \in X_{\min}$ and $\mathbf{y} + \mathbf{y}_2 \in \mathrm{cone}_{\mathbb{Z}}(Y_{\min})$. Since $(X_{\min}, Y_{\min})$ is minimal, $\mathbf{x} = \mathbf{x}_2$ and so, $\mathbf{y} + \mathbf{y}_2 = \mathbf{0}$. Finally, since C is pointed and $\mathbf{y}, \mathbf{y}_2 \in C$, $\mathbf{y} = \mathbf{0} = \mathbf{y}_2$, we deduce that $\mathbf{x} = \mathbf{x}'$, i.e. $\mathbf{x} \in X'$. We conclude that $X_{\min} \subseteq X'$, and similarly, one proves that $X' \subseteq X_{\min}$. So, $X' = X_{\min}$. □

Thanks to Theorem 21, a polyhedron P can be decomposed into a polytope Q and a cone C such that $P = Q + C$, and thanks to Theorem 22, C must be the characteristic cone of P. We show in the following theorem[1] that this decomposition leads to the possibility of generating all the integer elements of P.

[1]The proof of the existence of an extended Hilbert basis generating the integer elements of a polyhedron has been outlined in [Sch86], and the existence of a minimal extended Hilbert basis when the polyhedron is pointed has been mentioned without proof in [RV02].

Theorem 41. *For any convex polyhedron, there exists an extended Hilbert basis* (X, Y) *such that* $P \cap \mathbb{Z}^n = X + \text{cone}_{\mathbb{Z}}(Y)$ *and* $\text{char-cone}(P) \cap \mathbb{Z}^n = \text{cone}_{\mathbb{Z}}(Y)$.

If P *is pointed, then there exists a unique minimal extended Hilbert basis* $(X_{\min}, Y_{\min})$ *such that* $P \cap \mathbb{Z}^n = X_{\min} + \text{cone}_{\mathbb{Z}}(Y_{\min})$ *and* $\text{char-cone}(P) \cap \mathbb{Z}^n = \text{cone}_{\mathbb{Z}}(Y_{\min})$.

Proof. Thanks to Theorem 21, P can be decomposed into a polytope Q and a cone C such that $P = Q + C$, and thanks to Theorem 22, $C = \text{char-cone}(P)$. Also, thanks to Theorem 34, there exists an Hilbert basis $Y = \{\mathbf{y}_1, \ldots, \mathbf{y}_s\}$ such that $C \cap \mathbb{Z}^n = \text{cone}_{\mathbb{Z}}(Y)$. Based on the definition of a cone, one deduces that $C = \text{cone}_{\mathbb{Q}}(Y)$.

Let B be the polytope $B = \{\sum_{j=1}^{s} b_j \mathbf{y}_j \mid 0 \leq b_j \leq 1\}$.

We show that $P \cap \mathbb{Z}^n = \big((Q + B) \cap \mathbb{Z}^n\big) + \big(C \cap \mathbb{Z}^n\big)$ by proving the mutual inclusion.

- Let $\mathbf{x} \in (Q + B) \cap \mathbb{Z}^n$ and $\mathbf{y} \in C \cap \mathbb{Z}^n$, if $\mathbf{z} = \mathbf{x} + \mathbf{y}$, then $\mathbf{z} \in \mathbb{Z}^n$. Also, $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$ for some $\mathbf{x}_1 \in Q$ and $\mathbf{x}_2 \in B$. So, $\mathbf{z} = \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{y}$, i.e.

$$\mathbf{z} = \mathbf{x}_1 + \sum_{j=1}^{s} (b_j + b'_j)\mathbf{y}_j,$$

  such that $0 \leq b_j \leq 1$ and $b'_j \geq 0$. So, $\mathbf{z} = \mathbf{x}' + \mathbf{y}'$ such that $\mathbf{x}' \in Q$ and $\mathbf{y}' \in C$, and by definition, $\mathbf{z} \in P$. So, we conclude that

$$P \cap \mathbb{Z}^n \supseteq (Q + B) \cap \mathbb{Z}^n + C \cap \mathbb{Z}^n. \tag{3.4}$$

- Suppose $\mathbf{z} \in P \cap \mathbb{Z}^n$. By definition, $\mathbf{z} = \mathbf{x} + \mathbf{y}$ for some $\mathbf{x} \in Q$ and $\mathbf{y} \in C$. By definition, $\mathbf{y} = \sum_{j=1}^{s} b_j \mathbf{y}_j$. Let $\mathbf{y}_a, \mathbf{y}_b \in \mathbb{Q}^n$ such that

$$\begin{aligned} \mathbf{y}_a &= \sum_{j=1}^{s} \lfloor b_j \rfloor \mathbf{y}_j, \\ \mathbf{y}_b &= \mathbf{y} - \mathbf{y}_a. \end{aligned}$$

  By construction, we have $\mathbf{y}_a \in C \cap \mathbb{Z}^n$, $\mathbf{y}_b \in B$ and $\mathbf{z} = \mathbf{x} + \mathbf{y}_a + \mathbf{y}_b$. Finally, since $\mathbf{z} \in \mathbb{Z}^n$ and $\mathbf{y}_a \in \mathbb{Z}^n$, $\mathbf{x} + \mathbf{y}_b \in \mathbb{Z}^n$. So, $\mathbf{z} = \mathbf{x}' + \mathbf{y}'$ with $\mathbf{x}' \in (Q + B) \cap \mathbb{Z}^n$ and $\mathbf{y}' \in C \cap \mathbb{Z}^n$. We conclude that

$$P \cap \mathbb{Z}^n \subseteq (Q + B) \cap \mathbb{Z}^n + C \cap \mathbb{Z}^n. \tag{3.5}$$

Since both Q and B are polytopes, $(Q + B) \cap \mathbb{Z}^n$ is finite, i.e. there exists a finite set $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_t\} \subseteq \mathbb{Z}^n$ such that $(Q+B)\cap\mathbb{Z}^n = X$, and thus, we have

$$P \cap \mathbb{Z}^n = X + \mathrm{cone}_{\mathbb{Z}}(Y).$$

Suppose that P is pointed. By definition, $\mathrm{cone}_{\mathbb{Q}}(Y) = C = \text{char-cone}(P)$ is pointed, and the claim is a direct consequence of Lemma 40. □

Example 42. *The (minimal) extended Hilbert basis of the set $P \cap \mathbb{Z}^2$ with*

$$P = \{(x, y) \mid \begin{bmatrix} -4 & 1 \\ 1 & -1 \\ -1 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} -4 \\ 4 \\ -9 \end{bmatrix}\}$$

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

Chapter 4

Finite Automata

In this section, we introduce the notion of *finite automaton*, which is a finite data-structure used for representing potentially infinite sets of words. Two main properties of finite automata are that there is a canonical form, the minimum-state automaton, and set operations are easily performed on automata.

4.1 Basic Definitions

An *alphabet* is a (non-empty) finite set of symbols. A *word* over an alphabet Σ is a finite sequence of symbols taken from Σ. The symbol ε denotes the empty word, i.e. the word containing no symbol. The *length* of a word w, denoted by $|w|$, is the number of symbols in w. Given two words u and v, uv denotes the word formed by the concatenation of u with v. A word u is a *prefix* of a word w if there exists a word v such that $uv = w$. The set of prefixes of a word w is denoted by $\mathrm{pre}(w)$. A language L over Σ is a set of words over Σ. We denote by Σ^* (resp. Σ^+) the set of all (resp. non-empty) words over Σ.

Given two languages L_1 and L_2 over Σ, we define the *left-quotient* of L_1 with L_2, denoted as $L_2 \div L_1$, as the set of suffixes that complete words from L_2, such that the resulting word is in L_1. Formally, we have

$$L_2 \div L_1 = \{v \in \Sigma_r^* \mid uv \in L_1 \text{ for some } u \in L_2\}.$$

If $L_2 = \{w\}$, we simply write $w \div L_1$.

A *finite automaton (FA)* $\mathcal{A}$ is a quintuple $(Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$, where

- Q is a finite set of states,

- Σ is the input alphabet,
- $\Delta : Q \times \Sigma^* \times Q$ is the transition relation. For each transition $(q, w, q') \in \Delta$, q is the *origin*, w is the *label* and q' is the *destination*.
- $Q_{\mathtt{I}}$ is the set of initial states,
- $Q_{\mathtt{F}} \subseteq Q$ is the set of *final* states (also called *accepting* states).

If $(q, w, q') \in \Delta$ for $q, q' \in Q$ and $w \in \Sigma^*$, then we say that there is a *transition* from q to q' *labeled* by w. By extension, there is a *path* from q to q' *labeled* by w if there exists a sequence of transitions $(q_0, w_0, q_1), \ldots, (q_k, w_k, q_{k+1})$ such that $w = w_1 \cdots w_k$ and $q_0 = q$ and $q_{k+1} = q'$.

The set of words labeling paths from a state q_1 to a state q_2 in $\mathcal{A}$ is denoted as $L_{\mathcal{A}}(q_1 \to q_2)$. The *language accepted from a state* $q \in Q$, denoted $L_{\mathcal{A}}(q)$, is the set of words labeling paths in $\mathcal{A}$ from the state q to a final state. The *language accepted by* $\mathcal{A}$, denoted by $L(\mathcal{A})$, is the language accepted from the initial states. Two FAs are *equivalent* if they accept the same language. A language L is *regular* if there exists a FA $\mathcal{A}$ such that $L(\mathcal{A}) = L$.

An interesting feature of finite automata is that testing whether there is at least one word in the language accepted by a FA $\mathcal{A}$ can be done efficiently.

Proposition 43. *There exists an algorithm* AUTO_EMPTY? *which takes as arguments a FA,* $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ *and tests whether the language accepted by* $\mathcal{A}$ *is empty.*
The time cost of AUTO_EMPTY? *is* $\mathcal{O}(|\Delta|)$.

Proof. It suffices to test whether there is a path from an initial state to a final state, and this can be achieved through a depth first search [Tar72], as shown in Fig 4.1. □

A FA $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ is *reduced* if for all states $q \in Q$, q is reachable from an initial state and one can reach a final state from q, i.e. there exists an inital strate $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ and a final state $q_{\mathtt{F}} \in Q_{\mathtt{F}}$ such that $L_{\mathcal{A}}(q_{\mathtt{I}} \to q) \neq \emptyset$ and $L_{\mathcal{A}}(q \to q_{\mathtt{F}}) \neq \emptyset$.

Lemma 44. *There is a function* AUTO_REDUCE *which, given a FA* $\mathcal{A}$*, generates an equivalent reduced FA* $\mathcal{A}'$*. The time complexity of* AUTO_REDUCE *is* $\mathcal{O}(|\mathcal{A}|)$.

Proof. First one performs a backward search from the accepting state and remove all states from which there are no path towards an accepting state. Then one

```
function AUTO_EMPTY?(FA A = (Q, Σ_r^n, Δ, Q_I, Q_F)) : {false, true}
 1:   var Q_visited : set of state;
 2:           q : state;
 3:   function EXPLORE-FW(state q) : {false, true}
 4:       var q' : state;
 5:           w : word;
 6:       begin
 7:           Q_visited := Q_visited ∪ {q};
 8:           if q ∈ Q_F then return true;
 9:           for each (q, w, q') ∈ Δ, q' ∈ Q \ Q_visited do
10:               if EXPLORE-FW(q') then return true;
11:           return false;
12:       end
13:  begin
14:       for each q ∈ Q_I do
15:           if EXPLORE-FW(q) then return true;
16:       return false;
17:  end
```

Figure 4.1: Function AUTO_EMPTY?

performs a forward search from all initial states and remove all states which are not reachable. □

Let $\mathcal{A} = (Q, \sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ be a FA. If for each transition $(q, w, q') \in \Delta$, $|w| \leq 1$, then $\mathcal{A}$ is in *normal form*. If for each transition $(q, w, q') \in \Delta$, $|w| = 1$, then $\mathcal{A}$ is in *strong normal form*. The FA $\mathcal{A}$ is *deterministic* if

- $|Q_{\mathtt{I}}| = 1$,
- for all $(q, w, q') \in \Delta$, $w \neq \varepsilon$,
- for all distinct transitions $(q, w_1, q'_1), (q, w_2, q'_2) \in \Delta$, $w_1 \notin \mathrm{pre}(w_2)$ and $w_2 \notin \mathrm{pre}(w_1)$.

In the sequel, DFA stands for deterministic finite automaton.

Remark 45. *If a DFA is in normal form, by definition it is also in strong normal form.*

We have the following well-known results from automata theory.

Proposition 46. *There is an algorithm* AUTO_NORMALIZE *that takes a FA $\mathcal{A} = (Q, \Sigma, \Delta, Q_{init}, Q_{\mathtt{F}})$ as input and returns a FA in normal form $\mathcal{A}'$ such that $L(\mathcal{A}) = L(\mathcal{A}')$, and if $\mathcal{A}$ is deterministic, then $\mathcal{A}'$ is also deterministic.*

The time cost of AUTO_NORMALIZE *is $\mathcal{O}(|Q| + l)$, where l is the sum of the lengths of the labels of all transitions in $\mathcal{A}$, and the number of states in $\mathcal{A}'$ is $|Q| + l$.*

Proof. Let $\mathcal{A}' = (Q \cup Q', \Sigma, \Delta', Q_{init}, Q_{\mathtt{F}})$, where Q' and Δ' are computed from Q and Δ as follows. For each transition $(q, w, q') \in \Delta$,

- if $|w| \leq 1$, then one adds (q, w, q') to Δ',
- if $|w| > 1$, one adds $|w - 1|$ new states $q_1, \ldots, q_{|w|-1}$ to Q' and adds (q, α_1, q_1), (q_1, α_2, q_2), …, $(q_{|w|+1}, \alpha_{|w|}, q')$ to Δ' where $\alpha_i \in \Sigma$ for $i \in \{1, \ldots, n\}$ and $\alpha_1 \cdots \alpha_{|w|} = w$. □

Proposition 47. *There is an algorithm* AUTO_DETERMINIZE *that takes a FA $\mathcal{A} = (Q, \Sigma, \Delta, Q_{init}, Q_{\mathtt{F}})$ as input and returns a DFA in strong normal form $\mathcal{A}'$ such that $L(\mathcal{A}) = L(\mathcal{A}')$.*

The time cost of AUTO_DETERMINIZE *is $\mathcal{O}(2^{|Q|+|l|})$, where l is the sum of the lengths of the labels of all transitions in $\mathcal{A}$.*

Proof. First, one generates via the function AUTO_NORMALIZE($\mathcal{A}$) a FA $\mathcal{A}'' = (Q'', \Sigma, \Delta'', Q''_{\mathtt{I}}, Q''_{\mathtt{F}})$ in normal form, equivalent to $\mathcal{A}$. Thanks to Proposition 46, $|Q|'' \leq l+|Q|$. Then one constructs the finite automaton $\mathcal{A}' = (Q', \Sigma, \delta', \{q'_{\mathtt{I}}\}, Q'_{\mathtt{F}})$ such that

- Q' is the power set of Q'', i.e. the set of all subsets of Q'',
- $q'_{\mathtt{I}} = \{q \in Q'' \mid \varepsilon \in L_{\mathcal{A}''}(q' \to q) \text{ for some } q' \in Q''_{\mathtt{I}}\}$,
- Δ' is defined as follows. Given $S_1, S_2 \subseteq Q''$ and $\alpha \in \Sigma$, $(S_1, \alpha, S_2) \in \Delta'$ if $S_2 = \{q_2 \mid (\exists q_1 \in S_1)(\alpha \in L_{\mathcal{A}''}(q_1 \to q_2)\}$, that is, S_2 is exactly the sets of states reachable in $\mathcal{A}''$ from states in S_1 via paths labeled by α.
- $Q'_{\mathtt{F}}$ is the set of subsets Q'' containing at least one accepting state of $\mathcal{A}''$, i.e. $Q'_{\mathtt{F}} = \{S \subseteq Q'' \mid S \cap Q''_{\mathtt{F}} \neq \emptyset\}$.

By construction, $\mathcal{A}'$ is a DFA in normal form and $L(\mathcal{A}') = L(\mathcal{A})$. □

If a DFA $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ is in normal form, for each state q and for each symbol $\alpha \in \Sigma$, there is at most one state q' such that $(q, \alpha, q') \in \Delta$. Also, for any transition (q, w, q'), $|w| = 1$, i.e. $w \in \Sigma$. In order to emphasize those characteristics, the automaton will be described by the quintuple $(Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ where $\{q_{\mathtt{I}}\} = Q_{\mathtt{I}}$ and the partial function $\delta : Q \times \Sigma \to Q$ is deduced from the transition relation Δ by setting $\delta(q, \alpha) = q'$ if $(q, \alpha, q') \in \Delta$. If for all $q' \in Q$, $(q, \alpha, q') \notin \Delta$, we have $\delta(q, \alpha) = \perp$. Also, the partial function δ can be extended to words in the following way.

- for all $q \in Q$, $\hat{\delta}(q, \varepsilon) = q$,
- for all states $q \in Q$, words $w \in \Sigma^*$ and symbols $\alpha \in \Sigma$, $\hat{\delta}(q, \alpha w) = \begin{cases} \perp & \text{if } \delta(q, \alpha) = \perp \\ \hat{\delta}(\delta(q, \alpha), w) & \text{if } \delta(q, \alpha) \in Q \end{cases}$

Since $\hat{\delta}(q, \alpha) = \delta(q, \alpha)$ for all $\alpha \in \Sigma$, there can be no disagreement between δ and $\hat{\delta}$ on arguments for which both are defined.

A DFA in strong normal form $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ is *complete* if $\delta(q, \alpha) \in Q$ for all $q \in Q$ and $\alpha \in \Sigma$. Given a DFA in strong normal form $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$, we construct an equivalent complete DFA $\mathcal{A}'$ in strong normal form by adding a new state $q_\perp$ to Q and adding transitions labeled by $\alpha \in \Sigma$ from q to $q_\perp$ for all $q \in Q$ such that $\delta(q, \alpha) = \perp$. Note that a DFA in strong normal form can be

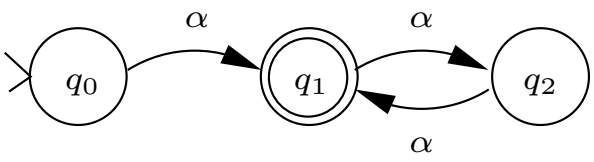


Figure 4.2: Example of FA which is not permutation-free

both reduced and complete. For example, the DFA $\mathcal{A} = (\{q_{\mathtt{I}}\}, \Sigma, \delta, q_{\mathtt{I}}, \{q_{\mathtt{I}}\})$, with $\delta(q_{\mathtt{I}}, \alpha) = q_{\mathtt{I}}$ for all $\alpha \in \Sigma$, accepting the language Σ^* is both complete and reduced.

Lemma 48. *Let $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be a DFA in strong normal form and $q \in Q$. If $\hat{\delta}(q_{\mathtt{I}}, w) = q$, then $w \div L(\mathcal{A}) = L_{\mathcal{A}}(q)$.*

Proof. This is a direct consequence of the definition of a DFA and of $w \div L$. Indeed, $wv \in L(\mathcal{A})$ iff $\hat{\delta}(q_{\mathtt{I}}, wv) \in Q_{\mathtt{F}}$, and since $\mathcal{A}$ is a DFA, $\hat{\delta}(q_{\mathtt{I}}, w) = q$. So, $wv \in L(\mathcal{A})$ iff $\hat{\delta}(q, v) \in Q_{\mathtt{F}}$, i.e. $v \in w \div L(\mathcal{A})$ iff $v \in L_{\mathcal{A}}(q)$. □

Finally, we consider two structural aspects of FAs.

A *strongly connected component* (SCC) $\mathcal{S}$ of a FA $\mathcal{A}$ is a pair (Q', Δ') such that

- $Q' \subseteq Q$,
- $\Delta' = \{(q, w, q') \in \Delta \mid q, q' \in Q'\}$, and,
- for all $q, q' \in Q'$, $L_{\mathcal{A}}(q \to q') \neq \emptyset$.

An SCC (Q', Δ') of $\mathcal{A}$ is *maximal* if for all $q \in Q \setminus Q'$ and $q' \in Q'$, either $L_{\mathcal{A}}(q \to q') = \emptyset$ or $L_{\mathcal{A}}(q' \to q) = \emptyset$.

Given a FA $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$, there is a (non-trivial) *permutation* of a subset of states $\{q_1, \ldots, q_k\} \subseteq Q$ *on* the word w if for $i \in \{1, \ldots, k\}$, $w \in L_{\mathcal{A}}(q_i \to q_{p(i)})$, where $p(1), \ldots, p(k)$ is a (non-trivial) permutation of $1, \ldots, k$. A FA is *permutation-free*[NP71] if there is no non-trivial permutation of any subset Q' on a word w. For example, the FA presented in Fig.4.2 is not permutation-free.

4.2 Minimal Automata

There is a well-known result in automata theory [Clu65, Har65, Hop71, HU79] stating that the minimum-state complete DFA $\mathcal{A}$ in strong normal form accepting

a regular language L is unique up to isomorphism, i.e. any complete DFA $\mathcal{A}'$ in strong normal form accepting L has more states than $\mathcal{A}$ or is equal to $\mathcal{A}$ up to renaming states. We give below a detailed proof of this result. The goal is twofold. First, in most algorithms in this thesis, we do not use minimum-state complete DFA but minimum-state reduced DFA, and the relationship between those are easily deduced from the detailed proof. Secondly, in many algorithms we use properties of minimum-state DFA that are clearly apparent from the detailed proof.

An equivalence relation R on Σ^* such that $(u, v) \in R$ implies that $(uw, vw) \in R$ for all words $w \in \Sigma^*$ is *right invariant* (with respect to concatenation).

Proposition 49. *Let R be a right invariant equivalence relation on Σ^* of finite index and L be the union of some equivalence classes of R.*

There exists a complete DFA $\mathcal{A}$ such that $L(\mathcal{A}) = L$ and such that the number of states in $\mathcal{A}$ is bounded by the number of equivalence classes of R.

Proof. Let $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ where

- $Q = \{[u]_R \mid u \in \Sigma^*\}$.
- For each $\alpha \in \Sigma$ and $[u]_R \in Q$, $\delta([u]_R, \alpha) = [u\alpha]_R$.
- $q_{\mathtt{I}} = [\varepsilon]_R$.
- $Q_{\mathtt{F}} = \{[u]_R \mid u \in L\}$.

By hypothesis, Q is finite and the definition of δ is consistent. Indeed, for all $u' \in [u]_R$, $(u, u') \in R$ and since R is right-invariant, for any $\alpha \in \Sigma$, $[u'\alpha]_R = [u\alpha]_R$.

For all $u \in \Sigma^*$ and $[u]_R \in Q$, we prove by induction on the size of u that $\hat{\delta}([\varepsilon]_R, u) = [u]_R$. This is trivially true for ε. Suppose that the property holds for words of length smaller or equal to k and $u = u_k\alpha$ with $\alpha \in \Sigma$, $u_k \in \Sigma^*$ and $|u_k| = k$. By inductive hypothesis, $\hat{\delta}([\varepsilon]_R, u_k) = [u_k]_R$, and by construction, $\delta([u_k]_R, \alpha) = [u_k\alpha]_R$.

From above, we conclude that $\mathcal{A}$ is a complete DFA with $L(\mathcal{A}) = L$. □

We may associate with an arbitrary language L a right invariant equivalence relation R_L on Σ^* such that for any word $u, v \in \Sigma^*$, $(u, v) \in R_L$ iff for each word $w \in \Sigma^*$, either $uw, vw \in L$ or $uw, vw \notin L$.

Proposition 50. *Let $\mathcal{A}(\Sigma, Q, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be a complete DFA and let $R_{\mathcal{A}}$ be the relation on Σ^* such that for all words $u, v \in \Sigma^*$, $(u, v) \in R_{\mathcal{A}}$ iff $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, v)$.*

- *The relation $R_{\mathcal{A}}$ is a right invariant equivalence relation and is a refinement of R_L, and*
- *$L(\mathcal{A})$ is a union of equivalence classes of $R_{\mathcal{A}}$.*

Proof.

- Clearly, $R_{\mathcal{A}}$ is an equivalence relation. In addition, by definition, for all words $u, v \in \Sigma^*$, if $(u, v) \in R_{\mathcal{A}}$, then $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, v)$. Therefore, for all words w, $\hat{\delta}(q_{\mathtt{I}}, uw) = \hat{\delta}(q_{\mathtt{I}}, vw)$ and $(uw, vw) \in R_{\mathcal{A}}$, i.e. $R_{\mathcal{A}}$ is right invariant. Also, for all words u, v, if $v \in [u]_{R_{\mathcal{A}}}$, $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, v)$, and therefore, for all words w, $\hat{\delta}(q_{\mathtt{I}}, uw) = \hat{\delta}(q_{\mathtt{I}}, vw)$, and so, $uw \in L$ iff vw, i.e. $v \in [u]_{R_L}$ and $R_{\mathcal{A}}$ is a refinement of R_L.
- By definition, $L(\mathcal{A}) = \cup_{q \in Q_{\mathtt{F}}} \{u \in \Sigma^* \mid \hat{\delta}(q_{\mathtt{I}}, u) = q\}$, and therefore, $L(\mathcal{A}) = \cup_{q \in Q_{\mathtt{F}}, u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)} [u]_{R_{\mathcal{A}}}$ □

Proposition 51 (The Myhill-Nerode Theorem, [HU79]). *Let $L \subseteq \Sigma^*$. The following three statements are equivalent.*

- *L is accepted by some finite automaton.*
- *L is the union of some equivalence classes of a right invariant equivalence relation of finite index.*
- *The equivalence relation R_L is of finite index and for any DFA $\mathcal{A}$ accepting L, $R_{\mathcal{A}}$ is a refinement of R_L.*

Proof. Direct consequence of Propositions 49 and 50. □

A direct consequence of the Myhill-Nerode Theorem is that for any regular language L, there exists a minimum-state complete DFA $\mathcal{A}$ accepting L unique up to isomorphism. That is, the number of states in $\mathcal{A}$ is smaller or equal to the number of states of any complete DFA $\mathcal{A}'$ accepting L, and if the equality holds, then $\mathcal{A}'$ is identical to $\mathcal{A}$ up to renaming of states. We call $\mathcal{A}$ the *complete minimal* DFA accepting L.

Proposition 52. *Given a regular language L, the complete minimal DFA $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ with $L(\mathcal{A}) = L$ is determined uniquely up to isomorphism.*

Proof. Let $\mathcal{A}' = (Q', \sigma, \delta', q'_{\mathtt{I}}, Q'_{\mathtt{F}})$ be a complete DFA with $L(\mathcal{A}') = L$.

Without loss of generality, we can assume that each state of $\mathcal{A}'$ is reachable since otherwise, the states that are not reachable can be removed without altering the language accepted by $\mathcal{A}$.

For each $q \in Q'_{\mathtt{F}}$, let $u_q \in \Sigma^*$ such that $\hat{\delta}'(q'_{\mathtt{I}}, u_q) = q$.

Since $\mathcal{A}'$ is complete, each equivalence class $[u]_{R_{\mathcal{A}'}}$ of the equivalence relation $R_{\mathcal{A}'}$ can be associated to one and only one state $q \in Q'$ such that

$$[u_q]_{R_{\mathcal{A}'}} = L_{\mathcal{A}'}(q_{\mathtt{I}} \to q).$$

Let $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be the complete DFA accepting L defined as in Proposition 49 when the right invariant equivalence relation is R_L. By construction, there is a bijection between the states q of $\mathcal{A}$ and the equivalence classes $[u_q]_{R_L}$ of R_L such that for all $v \in \Sigma^*$, $\hat{\delta}(q'_{\mathtt{I}}, v) = q$ iff $v \in [u_q]_{R_L}$.

Thanks to Proposition 50, $R_{\mathcal{A}'}$ is a refinement of R_L, and therefore, for each word $v \in \Sigma^*$, if $\delta'(q_{\mathtt{I}}, v) = q$ then $L_{\mathcal{A}'}(q_{\mathtt{I}} \to q) = [v]_{R_{\mathcal{A}'}} \subseteq [v]_{R_L}$. Since both $\mathcal{A}$ and $\mathcal{A}'$ are complete DFAs where all states are reachable from the initial state, we deduce that $|Q| \leq |Q'|$. If the equality holds, then each state $q' \in Q'$ can be identified with a state $q \in Q$ such that $L_{\mathcal{A}}(q_{\mathtt{I}} \to q) = L_{\mathcal{A}'}(q'_{\mathtt{I}} \to q')$, and therefore, $\mathcal{A}'$ is equal to $\mathcal{A}$ up to isomorphism. □

Let $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be a minimal complete DFA. From the proof above, we deduce the following properties of the minimal complete DFA. We first define a *sink state* q as a state such that $\delta(q, \alpha) = q$ for all $\alpha \in \Sigma$. A sink-state q is *accepting* if $q \in Q_{\mathtt{F}}$ and *non-accepting* otherwise.

Proposition 53. *For each state $q, q' \in Q$, $L_{\mathcal{A}}(q) \neq L_{\mathcal{A}}(q')$ iff $q \neq q'$.*

Proof. If $L_{\mathcal{A}}(q) \neq L_{\mathcal{A}}(q')$, then by definition, $q \neq q'$.

Conversely, if $L_{\mathcal{A}}(q) = L_{\mathcal{A}}(q')$, then $q = q'$ since otherwise, q and q' could be merged and $\mathcal{A}$ would not be minimal. □

Proposition 54. *If $\mathcal{A}$ is not reduced, then there exists one and only one non accepting sink state $q_\perp$. Also, for all $v \in \Sigma^*$ such that for all $w \in \Sigma^*$, $vw \notin L(\mathcal{A})$, $\delta(q_{\mathtt{I}}, v) = q_\perp$.*

Proof. Direct consequence of the proof of Proposition 52. □

Based on Proposition 53, there exists an algorithm which, when input a DFA, returns the equivalent minimal complete DFA.

Proposition 55. *There is an algorithm* AUTO_MINIMIZEC *that takes a DFA* $\mathcal{A} = (Q, \Sigma, \delta, Q_{init}, Q_{\mathtt{F}})$ *as input and returns the equivalent minimal complete DFA* $\mathcal{A}'$*. The time cost of* AUTO_MINIMIZEC *is* $\mathcal{O}(|Q| \log(|Q|))$.

Proof. Given in [Hop71]. □

In many algorithms presented in this thesis, we will require DFAs to be reduced. So, we define naturally the *reduced minimal* DFA accepting a regular language L as the minimum-state reduced DFA $\mathcal{A}_{\min}$ accepting L. We show that $\mathcal{A}_{\min}$ is the reduced DFA obtained from the complete minimal DFA $\mathcal{A}^c_{\min}$ essentially by removing the non-accepting sink state $q_\perp$ (if any) in $\mathcal{A}^c_{\min}$ as well as all transitions whose destination state is $q_\perp$.

First, note that given an alphabet Σ, the reduced minimal DFA accepting $\emptyset$ is $\mathcal{A}_{\min} = (\{q_{\mathtt{I}}\}, \Sigma, \delta, q_{\mathtt{I}}, \emptyset)$ with $\delta(q_{\mathtt{I}}, \alpha) = \emptyset$ for all $\alpha \in \Sigma$.

Proposition 56. *Let* $L \neq \emptyset$ *be a regular language and* $\mathcal{A}^c_{\min} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ *be the complete minimal DFA accepting* L*.*

- *If* $\mathcal{A}^c_{\min}$ *is reduced, then let* $\mathcal{A}_{\min} = \mathcal{A}^c_{\min}$*.*
- *If* $\mathcal{A}^c_{\min}$ *is not reduced, then let* $\mathcal{A}_{\min}$ *be the DFA obtained by removing the non-accepting sink state* $s_\perp$ *of* $\mathcal{A}^c_{\min}$ *as well as all transitions incoming in* $s_\perp$*.*

We have that $\mathcal{A}_{\min}$ *is a reduced DFA accepting* L *and for any DFA* $\mathcal{A}$ *accepting* L*,* $\mathcal{A}$ *has more states than* $\mathcal{A}_{\min}$ *or is equal to* $\mathcal{A}_{\min}$ *up isomorphism.*

Proof.

- Suppose $\mathcal{A}^c_{\min}$ is reduced. For all reduced DFAs $\mathcal{A}$ accepting L, without loss of generality, we can assume that all states of $\mathcal{A}$ are reachable and so, $\mathcal{A}$ must be complete (otherwise $L(\mathcal{A}) \neq L$), and therefore, thanks to Proposition 52, either $\mathcal{A}$ has more states than $\mathcal{A}_{\min}$ or they are equal up to isomorphism.
- Suppose $\mathcal{A}^c_{\min}$ is not reduced. Thanks to Proposition 54, there exists a non-accepting sink state and $\mathcal{A}_{\min}$ is well-defined. Also, by definition, $\mathcal{A}_{\min}$ is a reduced DFA.

  Let $\mathcal{A}$ be a reduced DFA accepting L. By definition, $\mathcal{A}$ is not complete but it can be completed by adding a new state $q_\perp$ and transitions labeled by $\alpha \in \Sigma$ from q to $q_\perp$ for all states q of $\mathcal{A}$ having no outgoing transitions labeled by

α. Let $\mathcal{A}^c$ be this complete DFA. Thanks to Proposition 52, either $\mathcal{A}^c$ has more states than $\mathcal{A}^c_{\min}$ or they are equal up to isomorphism. Consequently, $\mathcal{A}$ has more states than $\mathcal{A}_{\min}$ or they are equal up to isomorphism. □

From the above proposition and thanks to Proposition 55, we conclude this part with the following proposition.

Proposition 57. *There is an algorithm* AUTO_MINIMIZE *that takes a DFA* $\mathcal{A} = (Q, \Sigma, \delta, Q_{init}, Q_{\mathtt{F}})$ *as input and returns the equivalent minimal reduced DFA* $\mathcal{A}_{\min}$.

The time cost of AUTO_MINIMIZE *is* $\mathcal{O}(|Q| \log(|Q|))$.

Proof. It suffices to construct the complete minimal DFA $\mathcal{A}^c_{\min}$ accepting $L(\mathcal{A})$ and then to generate the reduced minimal DFA $\mathcal{A}_{\min}$ as in Proposition 56. □

4.3 Set Operations on Finite Automata

In this subsection, we briefly recall the cost of algorithms performing basic set operations on DFA. A description of the algorithms can be found in [HU79, Per90, Boi99]. Note that we assume that for each DFA, all states are reachable from the initial state.

Proposition 58. *There exists an algorithm* AUTO_PRODUCT *which takes as arguments two complete DFAs in normal form,* $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ *and* $\mathcal{A}_1 = (Q_2, \Sigma_2, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$*, and computes a complete DFA in normal form* $\mathcal{A}$ *over the alphabet* $\Sigma_1 \times \Sigma_2$ *accepting the language*

$$\{(w_1, w_2) \mid w_1 \in L(\mathcal{A}_1) \wedge w_2 \in L(\mathcal{A}_2) \wedge |w_1| = |w_2|\}.$$

The time cost of AUTO_PRODUCT *is* $\mathcal{O}(|Q_1| \cdot |Q_2| \cdot |\Sigma_1| \cdot |\Sigma_2|)$.

Proof. Construct the DFA $\mathcal{A} = (Q_1 \times Q_2, \Sigma_1 \times \Sigma_2, \delta, (q_{\mathtt{I},1}, q_{\mathtt{I},2}), Q_{\mathtt{F},1} \times Q_{\mathtt{F},2})$ where δ is such that $\delta\big(q_1, q_2), (\alpha_1, \alpha_2)\big) = (q'_1, q'_2)$ for each $q_1, q'_1 \in Q_1$, $q_2, q'_2 \in Q_2$, $\alpha_1 \in \Sigma_1$ and $\alpha_2 \in \Sigma_2$ such that $\delta_1(q_1, \alpha_1) = q'_1$ and $\delta_2(q_2, \alpha_2) = q'_2$. □

Proposition 59. *There exists an algorithm* AUTO_INTERSECTION *which takes as arguments two complete DFAs in normal form,* $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ *and* $\mathcal{A}_1 = (Q_2, \Sigma, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$*, and computes a complete DFA in normal form* $\mathcal{A}$ *accepting the language* $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$.
The time cost of AUTO_INTERSECTION *is* $\mathcal{O}(|Q_1| \cdot |Q_2| \cdot |\Sigma|)$.

Proof. Construct $\mathcal{A} = (Q_1 \times Q_2, \Sigma, \delta, (q_{\mathtt{I},1}, q_{\mathtt{I},2}), Q_{\mathtt{F},1} \times Q_{\mathtt{F},2})$ where δ is such that $\delta\big((q_1, q_2), \alpha\big) = (q_1', q_2')$ for each $q_1, q_1' \in Q_1$, $q_2, q_2' \in Q_2$, $\alpha \in \Sigma$ such that $\delta_1(q_1, \alpha) = q_1'$ and $\delta_2(q_2, \alpha) = q_2'$. □

Proposition 60. *There exists an algorithm* AUTO_UNION *which takes as arguments two complete DFAs in normal form, $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ and $\mathcal{A}_2 = (Q_2, \Sigma, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$, and computes a DFA $\mathcal{A}$ accepting the language $L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$.*
The time cost of AUTO_UNION *is $\mathcal{O}(|Q_1| \cdot |Q_2| \cdot |\Sigma|)$.*

Proof. Construct $\mathcal{A} = (Q_1 \times Q_2, \Sigma, \delta, (q_{\mathtt{I},1}, q_{\mathtt{I},2}), (Q_{\mathtt{F},1} \times Q_2) \cup (Q_1 \times Q_{\mathtt{F},2}))$, where δ is such that $\delta\big(q_1, q_2), \alpha\big) = (q_1', q_2')$ for each $q_1, q_1' \in Q_1$, $q_2, q_2' \in Q_2$, $\alpha \in \Sigma$ such that $\delta_1(q_1, \alpha) = q_1'$ and $\delta_2(q_2, \alpha) = q_2'$. □

Proposition 61. *There exists an algorithm* AUTO_COMPLEMENT *which takes as arguments a complete DFA in normal form, $\mathcal{A} = (Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ and computes a DFA $\mathcal{A}'$ over the alphabet Σ accepting the language $\Sigma^* \setminus L(\mathcal{A})$.*
The time cost of AUTO_COMPLEMENT *is $\mathcal{O}(|Q|)$.*

Proof. The complete DFA $\mathcal{A}'$ is $(Q, \Sigma, \delta, q_{\mathtt{I}}, Q \setminus Q_{\mathtt{F}})$. □

Proposition 62. *There exists an algorithm* AUTO_DIFFERENCE *which takes as arguments two complete DFAs in normal form, $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ and $\mathcal{A}_1 = (Q_2, \Sigma, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$, and computes a DFA $\mathcal{A}$ accepting the language $L(\mathcal{A}_1) \setminus L(\mathcal{A}_2)$.*
The time cost of AUTO_DIFFERENCE *is $\mathcal{O}(|Q_1| \cdot |Q_2| \cdot |\Sigma|)$.*

Proof. The complete DFA $\mathcal{A}$ is AUTO_INTERSECTION($\mathcal{A}_1$, $\mathcal{A}_3$), where $\mathcal{A}_3 =$ AUTO_COMPLEMENT$(\mathcal{A}_2)$. □

Proposition 63. *There exists an algorithm* AUTO_INCLUDED? *which takes as arguments two complete DFAs in normal form, $\mathcal{A}_1 = (Q_1, \Sigma, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ and $\mathcal{A}_1 = (Q_2, \Sigma, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$, and tests whether $L(\mathcal{A}_1) \subseteq L(\mathcal{A}_2)$.*
The time cost of AUTO_INCLUDED? *is $\mathcal{O}(|Q_1| \cdot |Q_2| \cdot |\Sigma|)$.*

Proof. This test can be done by checking the emptiness of the difference between $\mathcal{A}_1$ and $\mathcal{A}_2$. □

Proposition 64. *There exists an algorithm* AUTO_REVERSE *which takes as argument a FA* $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ *and computes a FA* $\mathcal{A}^R$ *such that for all words* $w \in \Sigma^*$, $w \in L(\mathcal{A})$ *iff* $w^R \in L(\mathcal{A}^R)$, *where* w^R *denotes the word* w *written from right to left.*
The time cost of AUTO_REVERSE *is* $\mathcal{O}(|\Delta| + l)$ *where* l *is the sum of the lengths of the labels of all transitions in* $\mathcal{A}$.

Proof. The FA $\mathcal{A}^R$ is $(Q, \Sigma, \Delta^R, Q_F, Q_{\mathtt{I}})$ such that $(q, w, q') \in \Delta$ iff $(q', w^R, q) \in \Delta^R$. □

The last operation presented in this section is the operation of homomorphism. A *homomorphism* is a function $f : \Sigma_1^* \to \Sigma_2^*$ such that for any two words $w_1, w_2 \in \Sigma_1^*$, we have $f(w_1 w_2) = f(w_1) f(w_2)$. According to this definition, a homomorphism is defined by the relation $\{(\alpha, f(\alpha)) \mid \alpha \in \Sigma\}$. Applying a homomorphism to an automaton $\mathcal{A}$ consists of computing an automaton $\mathcal{A}'$ such that $L(\mathcal{A}') = \{f(w) \mid w \in L(\mathcal{A})\}$. We have the following proposition.

Proposition 65. *There exists an algorithm* AUTO_HOMOMORPHISM *which takes as arguments one FA in strong normal form,* $\mathcal{A} = (Q, \Sigma, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ *and a homomorphism* $f : \Sigma^* \to \Sigma_f^*$, *and computes a FA* $\mathcal{A}_f$ *such that for all words* $w \in \Sigma^*$, $w \in L(\mathcal{A})$ *iff* $f(w) \in L(\mathcal{A}_f)$. *The time cost of* AUTO_HOMOMORPHISM *is* $\mathcal{O}(|\Delta|)$.

If for all $\alpha \in \Sigma$, $f(\alpha) \in \Sigma_f$, *then* $\mathcal{A}_f$ *is in strong normal form.*

Proof. We have $\mathcal{A}_f = (Q, \Sigma_f, \Delta_f, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ such that for all $(q, \alpha, q') \in \Delta$, we have $(q, f(\alpha), q') \in \Delta_f$, and there are no other transition in Δ_f. □

Chapter 5

Number Decision Diagrams

In this chapter, we explain how automata can represent particular sets of integer vectors. In addition, we give some properties of the proposed encoding scheme.

5.1 Automata-Based Representations of Integer Vector Sets

This idea of representing sets of integer by automata goes back at least to [Buc60] and it consists in establishing mappings between vectors and words, the mappings being based on the positional expression of numbers, with a signed-complement system for negative integers.

Given a positive integer number $r > 1$, any positive number z can be expressed as a sum of powers of r, $z = \sum_{i=0}^{p-1} a_i r^i$, with $a_i \in \{0, \ldots, r-1\}$, for some $p \in \mathbb{N}$. The sequence $a_{p-1}a_{p-2}\ldots a_0$ is then a word on the alphabet $\Sigma_r = \{0, \ldots, r-1\}$. The elements of Σ_r are called *digits*, and as defined, the encoding proposed is characterized by the fact that we start with the coefficient of the highest power of r in the decomposition, it is therefore called *most significant digit first (msdf)*. This encoding scheme is easily generalized to integer numbers by using the r-complement scheme according to which the encoding of a negative integer z with $-r^p \leq z < 0$ is given by the last $p+1$ digits of $r^{p+1} + z$.

Remark 66. *If $a_0, \ldots, a_p \in \{0, \ldots, r-1\}$ and $z \in \mathbb{Z}$, with $-r^p \leq z < 0$, satisfy the relation*

$$r^{p+1} + z = \sum_{i=0}^{p} a_i \cdot r^i,$$

then $r^{p+1} + z = (r-1) \cdot r^p + r^p - z$ *wih* $0 \leq r^p - z < r^p$*. Therefore, we have* $a_p = r - 1$ *and*

$$\begin{aligned} z &= -r^{p+1} + a_p \cdot r^p + \sum_{i=0}^{p-1} a_i \cdot r^i \\ &= -\frac{a_p}{r-1} \cdot r^p + \sum_{i=0}^{p-1} a_i \cdot r^i. \end{aligned}$$

If $a_0, \ldots, a_p \in \{0, \ldots, r-1\}$ *and* $z \in \mathbb{Z}$*, with* $0 \leq z < r^p$*, satisfy the relation*

$$z = \sum_{i=0}^{p} a_i \cdot r^i,$$

then $a_p = 0$ *and*

$$\begin{aligned} z &= a_p \cdot r^p + \sum_{i=0}^{p-1} a_i \cdot r^i \\ &= -\frac{a_p}{r-1} \cdot r^p + \sum_{i=0}^{p-1} a_i \cdot r^i. \end{aligned}$$

□

Formally, we have the following definition.

Definition 67. *Given an* encoding basis $r > 1$*, a word* $w = a_p a_{p-1} \ldots a_0$ *with* $p \in \mathbb{N}$*,* $a_p \in \{0, r-1\}$ *and* $a_i = \{0, \ldots, r-1\}$ *for* $i \in \{0, \ldots, p-1\}$*, is an* msdf r-encoding of an integer $z \in \mathbb{Z}$*, denoted by* $\langle w \rangle_r = z$*, if*

$$z = -\frac{a_p}{r-1} \cdot r^p + \sum_{i=0}^{p-1} a_i r^i.$$

□

For example, the words 0120 and 2201 are 3-encodings of the numbers 15 and -8 respectively.

Note that a word $w \in \Sigma_r^*$ is an r-encoding of some integer z if and only if $|w| \geq 1$ and the first symbol a of w belongs to $\{0, r-1\}$. If $a = r - 1$, then $z < 0$, and conversely, if $a = 0$, $z \geq 0$. The first digit of w is therefore called the *sign digit*. Also, two r-encodings of the same number differ only by the repetitions of the sign digit, as proved in the following lemma.

Lemma 68. *Let $w_1, w_2 \in (\Sigma_r)^+$ be r-encodings.*

There exist $a \in \{0, r-1\}$ and $w \in \Sigma_r^*$ such that $w_1 = a^{k_1}w$ and $w_2 = a^{k_2}w$ for some $k_1, k_2 \in \mathbb{N} \setminus \{0\}$ if and only if w_1 and w_2 are r-encodings of the same number.

Proof.

- Suppose that $w_1, w_2 \in \Sigma_r^+$ are r-encodings of some $z \in \mathbb{Z}$. Without loss of generality, assume that $|w_1| \leq |w_2|$, and let $a_{1,|w_1|-1}, a_{2,|w_2|-1} \in \{0, r-1\}$ and $a_{1,0}, \ldots, a_{1,|w_1|-2}, a_{2,0}, \ldots, a_{2,|w_2|-2} \in \{0, \ldots, r-1\}$ such that $w_1 = a_{1,|w_1|-1} \ldots a_{1,0}$ and $w_2 = a_{2,|w_2|-1} \ldots a_{2,0}$. By definition of the encoding scheme, we have

$$\langle w_1 \rangle_r = -r^{|w_1|-1}\frac{a_{1,|w_1|-1}}{r-1} + \sum_{i=0}^{|w_1|-2} a_{1,i}r^i \tag{5.1}$$

$$\langle w_2 \rangle_r = -r^{|w_2|-1}\frac{a_{2,|w_2|-1}}{r-1} + \sum_{i=0}^{|w_2|-2} a_{2,i}r^i. \tag{5.2}$$

  The sign digits of w_1 and w_2 must be equal and so, $a_{1,|w_1|-1} = a_{2,|w_2|-1} = a$. Since $\langle w_1 \rangle_r = z = \langle w_2 \rangle_r$, we deduce that

$$0 = -r^{|w_2|-1}\frac{a}{r-1} + \sum_{i=|w_1|}^{|w_2|-2} a_{2,i}r^i + (a_{2,|w_1|-1} + \frac{a}{r-1})r^{|w_1|-1} + \sum_{i=0}^{|w_1|-2} (a_{2,i} - a_{1,i})r^i. \tag{5.3}$$

  So, $a_{2,i} = a_{1,i}$ for $i \in \{0, \ldots, |w_1| - 2\}$ and

$$-r^{|w_2|-1}\frac{a}{r-1} + \sum_{i=|w_1|-1}^{|w_2|-2} a_{2,i}r^i = -r^{|w_1|-1}\frac{a}{r-1}. \tag{5.4}$$

  If $a = 0$, then $\sum_{i=|w_1|-1}^{|w_2|-2} a_{2,i}r^i = 0$, and so $a_{2,i} = 0$ for $i \in \{|w_1|, \ldots, |w_2| - 1\}$. Conversely, if $a = r-1$, then $\sum_{i=|w_1|-1}^{|w_2|-2} a_{2,i}r^i = r^{|w_2|} - r^{|w_1|}$. Since $a_{2,i} \leq r-1$, for all $i \in \{|w_1| - 1, \ldots, |w_2| - 2\}$, $\sum_{i=|w_1|-1}^{|w_2|-2} a_{2,i}r^i = r^{|w_2|} - r^{|w_1|}$ iff $a_{2,i} = r-1$ for all $i \in \{|w_1| - 1, \ldots, |w_2| - 2\}$.

- Suppose that there exist $a \in \{0, r-1\}$ and $w \in \Sigma_r^*$ such that $w_1 = a^{k_1}w$ and $w_2 = a^{k_2}w$ for some $k_1, k_2 \in \mathbb{N} \setminus \{0\}$. We show that for any k,

$$\langle a^k \rangle_r = \langle a \rangle_r.$$

Once this is proved, by definition of the encoding scheme, we deduce that $\langle a^k w\rangle_r = \langle aw\rangle_r$, and therefore we have

$$\langle a^{k_1} w\rangle_r = \langle a^{k_2} w\rangle_r.$$

Clearly, if $k = 1$, then $\langle a^k\rangle_r = \langle a\rangle_r$. If $k \geq 2$, by definition of the encoding scheme, we have

$$\begin{aligned}
\langle a^k\rangle_r &= -r^{k-1}\frac{a}{r-1} + \sum_{i=0}^{k-2} r^i a \\
&= -\frac{r^{k-1}}{r-1}a + \frac{r^{k-1}-1}{r-1}a \\
&= -\frac{a}{r-1} \\
&= \langle a\rangle_r.
\end{aligned}$$

□

Definition 69. *The r-encoding of $z \in \mathbb{Z}$ having no repetition of the sign digit is called the* minimal r-encoding of z. □

Note that thanks to Lemma 68, given an integer $z \in \mathbb{Z}$, the minimal r-encoding of z is unique.

In order to encode a vector $\mathbf{z} \in \mathbb{Z}^n$, it suffices to read synchronously one digit from the encodings of all its components, provided that these encodings share the same length. This requirement can always be met by prefixing the minimal r-encodings of the components by a sequence of copies of their sign digit. An r-encoding of a vector $\mathbf{z} \in \mathbb{Z}^n$ can indifferently be viewed as a tuple of n words in $(\Sigma_r)^*$ or as a word in $(\Sigma_r^n)^*$. In the following, we adopt the latter, and therefore, each symbol $\alpha \in \Sigma_r^n$ is a tuple of digits $(a_1, \ldots, a_n)$ with $a_i \in \{0, \ldots, r-1\}$. For convenience, the ith element of α is denoted $\alpha[i]$.

Formally, we have the following definition.

Definition 70. *Given a dimension $n \geq 0$ and an encoding basis $r > 1$, a word $w = \alpha_p\alpha_{p-1}\ldots\alpha_0$ over Σ_r^n, with $p \in \mathbb{N}$ and $p \geq 1$, is a* (msdf) r-encoding of an integer vector $\mathbf{z} \in \mathbb{Z}^n$, *denoted $\langle w\rangle_{r,n} = \mathbf{z}$ if for each $j \in \{1, \ldots, n\}$, $\langle \alpha_p[j]\alpha_{p-1}[j]\ldots\alpha_0[j]\rangle_r = \mathbf{z}[j]$.*

The synchronous encoding scheme $E_{S(r)}$ *is the relation that associates to a vector $\mathbf{z} \in \mathbb{Z}^n$ the words $w \in (\Sigma_r^n)^*$ such that $\mathbf{z} = \langle w\rangle_{r,n}$.* □

Note that a word $w \in (\Sigma_r^n)^*$ is an r-encoding of some integer vector if and only if $|w| \geq 1$ and the first symbol of w belongs to $\{0, r-1\}^n$. Generalizing the notion of sign digit, the first symbol of any r-encoding is called the *sign symbol*. Since sets of integer vectors whose components are all positive will play an important role in the sequel, for convenience, we will use the symbol o to denote $(0, \ldots, 0)$.

Lemma 68 easily generalizes to the vector case as follows.

Lemma 71. *Let $w_1, w_2 \in (\Sigma_r^n)^+$ be r-encodings.*

There exist $\alpha \in \{0, r-1\}^n$ and $w \in (\Sigma_r^n)^*$ such that $w_1 = \alpha^{k_1} w$ and $w_2 = \alpha^{k_2} w$ for some $k_1, k_2 \in \mathbb{N} \setminus \{0\}$ iff w_1 and w_2 are r-encodings of the same vector.

Proof. Direct consequence of the encoding scheme and of Lemma 68. □

As in the case of a single integer, we deduce from the above lemma the existence of an r-encoding of an integer vector with minimal length.

Definition 72. *The encoding of $\mathbf{z} \in \mathbb{Z}^n$ having no repetition of the sign symbol is called the* minimal r-encoding of $\mathbf{z}$. □

Note that thanks to Lemma 71, given an integer vector $\mathbf{z} \in \mathbb{Z}$, the minimal r-encoding of $\mathbf{z}$ is unique. Also, the sign symbols of all encodings of a vector are identical. We define the function $\mathrm{sign}_r : \mathbb{Z}^n \to \Sigma_R^n$ such that $\mathrm{sign}_r(\mathbf{z})$ returns the sign symbol of the r-encodings of $\mathbf{z}$.

Remark 73. *By definition, for all $\mathbf{x} \in \mathbb{Z}^n$, we have $\mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}$ iff for all $i \in \{1, \ldots n\}$, $\begin{cases} \mathbf{x}[i] < 0 \text{ if } \alpha_{\mathtt{sign}}[i] = r-1 \\ \mathbf{x}[i] \geq 0 \text{ if } \alpha_{\mathtt{sign}}[i] = 0 \end{cases}$.* □

Let $S \subseteq \mathbb{Z}^n$ with $n \geq 1$. If the language $L(S)$ containing all the r-encodings of all the vectors in S is regular, then any finite automaton $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ accepting $L(S)$, i.e. such that $L(\mathcal{A}) = L(S)$, is a *Number Decision Diagram (NDD)*, and we say that $\mathcal{A}$ *represents* S and S is *recognizable* with respect to the synchronous encoding scheme $E_{S(r)}$. In this paper, we use the following notations. We denote by $S_{\mathcal{A}}^q$ the set of integer vectors whose encodings label paths from any $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ to q in the NDD $\mathcal{A}$, and by $S_{\mathcal{A}}$ the set of integer vectors represented by the NDD $\mathcal{A}$.

The choice of accepting all the encodings of the elements in the set is based on the fact that most set operations on sets recognizable with respect to $E_{S(r)}$ can be performed by applying the corresponding operations on the automata. There is only some extra cost for the projection operation [Boi99, BL01].

Automata-based representations of sets of integer vectors have been studied for a long time, at least back to [Buc60]. We conclude this section by providing

two important results regarding recognizable sets of integer vectors. The first theorem characterizes the sets of integer vectors that are recognizable with respect to $E_{S(r)}$, for some $r > 1$, the second characterizes the sets that are recognizable with respect to $E_{S(r)}$, for all $r > 1$.

Theorem 74. *The sets definable in the first order theory $\langle \mathbb{Z}, 0, 1, +, V_r, < \rangle$, where V_r is the function defined as follows.*

$$V_r : \mathbb{Z} \to \mathbb{Z} : z \mapsto \begin{cases} \textit{the greatest power of } r \textit{ dividing } z & \textit{if } z \neq 0, \\ 1 & \textit{if } z = 0. \end{cases}$$

correspond exactly to the sets that are recognizable with respect to the synchronous encoding scheme $E_{S(r)}$, $r > 1$.

Proof. [Buc60, McN63, Bru85, MP86, Vil92, Boi99] □

Theorem 75. *The sets definable in the first order theory $\langle \mathbb{Z}, 0, 1, +, < \rangle$ correspond exactly to the sets that are recognizable with respect to all synchronous encoding schemes $E_{S(r)}$ with $r > 1$.*

Proof. [Cob69, Sem77, BHMV94, MV96, Muc03][1]. □

Finally, we give the bound on the size of the minimal NDD representing a set defined by a Presburger formula φ.

Theorem 76. *For any Presburger formula $\varphi(x_1, \ldots, x_n)$, there exists a complete minimal NDD $\mathcal{A}$ accepting the (msdf) r-encodings of the elements of the set $S = \{(x_1, \ldots, x_n) \in \mathbb{Z}^n \mid \varphi(x_1, \ldots, x_n)\}$ with at most $2^{2^{2^{|\varphi|}}}$ states.*

Proof. [Kla04b, Kla04a]. □

5.2 Basic Operations on NDDs

In this section, we detail procedures for performing basic operations on NDDs. The operations of intersection, union, complement, difference and product can be directly performed on the corresponding operations on automata. The procedure corresponding to the projection is similar to the operation of homomorphism but required some additional considerations.

[1] The proofs actually apply to subsets of $\mathbb{N}^n$. The generalization to subsets of $\mathbb{Z}^n$ is immediate.

Theorem 77. *Let $\mathcal{A}_1 = (Q_1, \Sigma_r^n, \delta_1, q_{\mathtt{I},1}, Q_{\mathtt{F},1})$ and $\mathcal{A}_2 = (Q_2, \Sigma_r^n, \delta_2, q_{\mathtt{I},2}, Q_{\mathtt{F},2})$ be deterministic NDDs in strong normal form representing the set $S_1 \subseteq \mathbb{Z}^n$ and $S_2 \subseteq \mathbb{Z}^n$.*

The DFAs in strong normal form AUTO_INTERSECTION*($\mathcal{A}_1$, $\mathcal{A}_2$),* AUTO_UNION*($\mathcal{A}_1$, $\mathcal{A}_2$),* AUTO_DIFFERENCE*($\mathcal{A}_1$, $\mathcal{A}_2$) and* AUTO_PRODUCT*($\mathcal{A}_1$, $\mathcal{A}_2$) are deterministic NDD in normal form representing the sets $S_1 \cap S_2$, $S_1 \cup S_2$, $S_1 \setminus S_2$ and $S_1 \times S_2$.*

Proof. Direct consequence of the fact that all encodings of the elements in a set S are accepted by an NDD representing S. □

Theorem 78. *There exists a procedure* NDD_COMPLEMENT *which, given a NDD in normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ representing the set $S \subseteq \mathbb{Z}^n$, generates a deterministic NDD $\mathcal{A}'$ in strong normal form representing $\mathbb{Z}^n \setminus S$.*

The time complexity of NDD_COMPLEMENT *is $\mathcal{O}(2^{|Q|} \cdot |\Sigma_r^n|)$ and the number of states in $\mathcal{A}'$ is at most $\mathcal{O}(2^{|Q|})$.*

Proof. Let $\mathcal{A}_{\mathbb{Z}^n} = (\{q_1, q_2\}, \Sigma_r^n, \delta_{\mathbb{Z}^n}, q_1, \{q_2\})$ such that $\delta(q_1, \alpha) = q_2$ for all $\alpha \in \{0, r-1\}^n$ and $\delta(q_2, \alpha) = q_2$ for all $\alpha \in \Sigma_r^n$. By construction, $\mathcal{A}_{\mathbb{Z}^n}$ is the minimal NDD representing $\mathbb{Z}^n$.

The NDD $\mathcal{A}'$ is AUTO_DETERMINIZE(AUTO_DIFFERENCE($\mathcal{A}_{\mathbb{Z}^n}$, $\mathcal{A}$)). □

Finally, we define the *projection* operation as follows. Given a set $S \subseteq \mathbb{Z}^n$, $n \geq 2$, the projection of S with respect to the ith component, denoted $\exists_i(S)$, is defined as follows.

$$\exists_i(S) = \{(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) \mid (x_1, \ldots, x_n) \in S\}.$$

Theorem 79. *There exists a procedure* NDD_PROJECTION *which, given an integer $i \in \{1, \ldots, n\}$ and an NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ representing the set $S \subseteq \mathbb{Z}^n$, $n \geq 2$, generates an NDD $\mathcal{A}'$ in strong normal form representing $\exists_i(S)$.*

The time complexity of NDD_PROJECTION *is $\mathcal{O}(|\Delta| \cdot |\Sigma_r^n|)$ and the number of states in $\mathcal{A}'$ is $|Q|$.*

Proof. Let f be the homomorphism mapping Σ_r^n onto Σ_r^{n-1} by removing the ith component. Formally, $f : \Sigma_r^n \to \Sigma_r^{n-1}$ with

$$\begin{cases} \varepsilon \to \varepsilon \\ \alpha \to (\alpha[1], \ldots, \alpha[i-1], \alpha[i+1], \ldots, \alpha[n] \\ w_1 w_2 \to f(w_1) f(w_2) \end{cases}$$

Let $\mathcal{A}_f$ be the FA generated via the call AUTO_HOMOMORPHISM($\mathcal{A}$, f). Thanks to Lemma 65, $\mathcal{A}_f$ is in strong normal form.

For each $w \in L(\mathcal{A}_f)$, $\langle w \rangle_{r,n-1} \in \exists_i(S)$. However, if $\mathbf{x} \in S$, then some encodings of $\mathbf{x}$ might not be accepted by $\mathcal{A}_f$. For example, if $S = \{(0,3)\}$, then $L(\mathcal{A}) = \{(0,0)^k(0,1)(0,1) \mid k \geq 1\}$. By definition, $\exists_1(S) = \{(0)\}$ and $L(\mathcal{A}_f) = \{(0)^k(0)(0) \mid k \geq 1\}$.

In order to add the missing encodings of elements of $\exists_i(S)$, for each encodings α^k, $k \geq 1$ labeling a path from an initial state to a state q in $\mathcal{A}_f$, one needs to add the paths labeled by $\alpha^{k'}$ for all $k' \geq 1$. This can be done by performing for each $\alpha \in \{O, r-1\}^{n-1}$ a depth-first-search starting at the states in $Q_{\mathtt{I}}$, and following only transitions labeled by α. A formal description of NDD_PROJECTION is given in Fig 5.1. □

The function NDD_PROJECTION is easily generalized in order to project with respect to a set of components. Given a set $S \subseteq \mathbb{Z}^n$, $n \geq 2$, the projection of S with respect to a set of component I, denoted $\exists_I(S)$ is defined recursively as follows.

$$\begin{aligned} \exists_{\{i\}}(S) &= \exists_i(S) \\ \exists_{\{i_1,\ldots,i_{k+1}\}}(S) &= \exists_{\{i_1,\ldots,i_k\}}\left(\exists_{i_{k+1}}(S)\right) \end{aligned}$$

Theorem 80. *There exists a procedure* NDD_MULTI_PROJECTION *which, given a set $I \subseteq \{1,\ldots,n\}$ and an NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ representing the set $S \subseteq \mathbb{Z}^n$, $n \geq 1$, generates an NDD $\mathcal{A}'$ in strong normal form representing $\exists_I(S)$.*

The time complexity of NDD_MULTI_PROJECTION *is $\mathcal{O}(|\Delta| \cdot |\Sigma_r^n|)$ and the number of states in $\mathcal{A}'$ is $|Q|$.*

Proof. The proof of Theorem 79 is easily adapted for the projection with respect to a set of components. □

5.3 Synchronous Encoding Scheme and Linear Constraints

In this section, we describe some properties of the encoding scheme. Based on those properties, we present in Section 5.4 an efficient algorithm for generating an NDD for a set given a quantifier-free Presburger formula defining the set, which is

function NDD_PROJECTION(integer i, NDD $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_\mathtt{I}, Q_\mathtt{F})$) : NDD

1: **var** $Q_\mathtt{visited}$: **set of** state;
2: q, q', q'' : state;
3: f : function;
4: $(Q', \Sigma', \Delta', Q'_\mathtt{I}, Q'_\mathtt{F})$: automaton;
5: α, α' : symbol;
6: **procedure** EXPLORE-FW(state s)
7: **var** s' : state;
8: **begin**
9: $Q_\mathtt{visited} := Q_\mathtt{visited} \cup \{s\}$;
10: **for each** $(s, \alpha', s') \in \Delta'$ **do**
11: **begin**
12: $\Delta' := \Delta' \cup \{(q', \alpha', s'), (q'', \alpha', s')\}$;
13: **if** $s' \notin Q_\mathtt{visited}$ **then** EXPLORE-FW(s');
14: **end**
15: **end**
16: **begin**
17: **let** $f : \Sigma_r^n \to \Sigma_r^{n-1} : \begin{cases} \varepsilon \to \varepsilon \\ \alpha \to (\alpha[1], \ldots, \alpha[i-1], \alpha[i+1], \ldots, \alpha[n]; \\ w_1 w_2 \to f(w_1) f(w_2) \end{cases}$
18: $(Q', \Sigma', \Delta', Q'_\mathtt{I}, Q'_\mathtt{F})$:= AUTO_HOMOMORPHISM($\mathcal{A}$, f);
19: **let** $q' \notin Q'$;
20: $Q' := Q' \cup \{q'\}$;
21: $Q'_\mathtt{I} := Q'_\mathtt{I} \cup \{q'\}$;
22: **for each** $\alpha \in \{0, r-1\}^{n-1}$ **do**
23: **begin**
24: **let** $q'' \notin Q'$;
25: $Q' := Q' \cup \{q''\}$;
26: $\Delta' := \Delta' \cup \{(q', \alpha, q''), (q'', \alpha, q'')\}$;
27: $Q_\mathtt{visited} := \emptyset$;
28: **for each** $q \in Q_\mathtt{I}$ **do** EXPLORE-FW(q);
29: **end**
30: **return** $(Q', \Sigma', \Delta', Q'_\mathtt{I}, Q'_\mathtt{F})$;
31: **end**

Figure 5.1: Function NDD_PROJECTION

essentially the one given in [Kla04b, Kla04a]. The properties are also used when analyzing the structure of NDDs.

Intuitively, the following lemmas show how, given an encoding $u \in (\Sigma_r^n)^+$ and a vector $\mathbf{a} \in \mathbb{Z}^n$, the value of $\mathbf{a}.\langle u \rangle_{r,n}$ constrains the possible values of $\mathbf{a}.\langle uv \rangle_{r,n}$ for any word $v \in (\Sigma_r^n)^*$.

Lemma 81. *Let $\mathbf{a} \in \mathbb{Z}^n$ and u_1, u_2 be encodings over Σ_r^n.*

1. *If $\mathbf{a}.\langle u_1 \rangle_{r,n} \leq \mathbf{a}.\langle u_2 \rangle_{r,n}$, then for all words $v \in (\Sigma_r^n)^*$, we have*

$$\mathbf{a}.\langle u_1 v \rangle_{r,n} \leq \mathbf{a}.\langle u_2 v \rangle_{r,n}.$$

2. *If $\mathbf{a}.\langle u_1 \rangle_{r,n} \equiv_m \mathbf{a}.\langle u_2 \rangle_{r,n}$, then for all words $v \in (\Sigma_r^n)^*$, we have*

$$\mathbf{a}.\langle u_1 v \rangle_{r,n} \equiv_m \mathbf{a}.\langle u_2 v \rangle_{r,n}.$$

Proof. By definition of the encoding scheme, $\langle uv \rangle_{r,n} = r^{|v|}\langle u \rangle_{r,n} + \langle ov \rangle_{r,n}$.

1. If $\mathbf{a}.\langle u_1 \rangle_{r,n} \leq \mathbf{a}.\langle u_2 \rangle_{r,n}$, then we have

$$\begin{aligned} \mathbf{a}.\langle u_1 v \rangle_{r,n} &= \mathbf{a}.\left(r^{|v|}\langle u_1 \rangle_{r,n} + \langle ov \rangle_{r,n}\right) \\ &\leq \mathbf{a}.\left(r^{|v|}\langle u_2 \rangle_{r,n} + \langle ov \rangle_{r,n}\right) \\ &\leq \mathbf{a}.\langle u_2 v \rangle_{r,n}\ . \end{aligned}$$

2. If $\mathbf{a}.\langle u_1 \rangle_{r,n} \equiv_m \mathbf{a}.\langle u_2 \rangle_{r,n}$, then we have

$$\begin{aligned} \mathbf{a}.\langle u_1 v \rangle_{r,n} &\equiv_m \mathbf{a}.\left(r^{|v|}\langle u_1 \rangle_{r,n} + \langle ov \rangle_{r,n}\right) \\ &\equiv_m \mathbf{a}.\left(r^{|v|}\langle u_2 \rangle_{r,n} + \langle ov \rangle_{r,n}\right) \\ &\equiv_m \mathbf{a}.\langle u_2 v \rangle_{r,n}\ . \end{aligned}$$

□

Lemma 82. *Let $\mathbf{a} \in \mathbb{Z}^n$ and let $u \in (\Sigma_r^n)^+$ be an encoding.*

1. *If $\mathbf{a}.\langle u \rangle_{r,n} \leq -\| \mathbf{a}^+ \|$, then $\forall v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv \rangle_{r,n} \leq \mathbf{a}.\langle u \rangle_{r,n}$.*
2. *If $\mathbf{a}.\langle u \rangle_{r,n} < -\| \mathbf{a}^+ \|$, then $\forall v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv \rangle_{r,n} < \mathbf{a}.\langle u \rangle_{r,n}$.*
3. *If $\mathbf{a}.\langle u \rangle_{r,n} \geq \| \mathbf{a}^- \|$, then $\forall v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv \rangle_{r,n} \geq \mathbf{a}.\langle u \rangle_{r,n}$.*
4. *If $\mathbf{a}.\langle u \rangle_{r,n} > \| \mathbf{a}^- \|$, then $\forall v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv \rangle_{r,n} > \mathbf{a}.\langle u \rangle_{r,n}$.*

Proof. We prove (1), the proofs for (2), (3) and (4) are similar.

Suppose that $\mathbf{a}.\langle u\rangle_{r,n} \leq -\|\mathbf{a}^+\|$. By definition of the encoding scheme, $\langle uv\rangle_{r,n} = r^{|v|}\langle u\rangle_{r,n} + \langle ov\rangle_{r,n}$, and $0 \leq \langle ov\rangle_{r,n}[i] \leq r^{|v|} - 1$, for $i \in \{1,\ldots,n\}$, implying that $0 \leq \mathbf{a}.\langle 0v\rangle_{r,n} \leq (r^{|v|}-1)\|\mathbf{a}^+\|$. So, we have

$$\begin{aligned}
\mathbf{a}.\langle uv\rangle_{r,n} &= \mathbf{a}.(r^{|v|}\langle u\rangle_{r,n} + \langle ov\rangle_{r,n}) \\
&= \mathbf{a}.\langle u\rangle_{r,n} + (r^{|v|}-1)\mathbf{a}.\langle u\rangle_{r,n} + \mathbf{a}.\langle ov\rangle_{r,n} \\
&\leq \mathbf{a}.\langle u\rangle_{r,n} - (r^{|v|}-1)\|\mathbf{a}^+\| + (r^{|v|}-1)\|\mathbf{a}^+\| \\
&\leq \mathbf{a}.\langle u\rangle_{r,n}.
\end{aligned}$$

□

Lemma 83. *Let* $\mathbf{a}.\mathbf{x} \boxtimes b$ *where* $\boxtimes \in \{<,\leq,=,\geq,>\}$. *Let* $u \in (\Sigma_r^n)^+$ *be an encoding.*

1. *If* $\mathbf{a}.\langle u\rangle_{r,n} < \min(b, -\|\mathbf{a}^+\|)$, *then for all* $v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv\rangle_{r,n} \boxtimes b$ *holds iff* $\boxtimes \in \{<,\leq\}$.

2. *If* $\mathbf{a}.\langle u\rangle_{r,n} > \max(b, \|\mathbf{a}^-\|)$, *then for all* $v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv\rangle_{r,n} \boxtimes b$ *holds iff* $\boxtimes \in \{>,\geq\}$.

Proof. We prove (1), the proof for (2) is similar.

Suppose that $\mathbf{a}.\langle u\rangle_{r,n} < \min(b, -\|\mathbf{a}^+\|)$. Thanks to Lemma 82, for all $v \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uv\rangle_{r,n} < \mathbf{a}.\langle u\rangle_{r,n} < b$. So, $\mathbf{a}.\langle uv\rangle_{r,n} \boxtimes b$ holds iff $\boxtimes \in \{<,\leq\}$. □

Thanks to the above lemma, given an (in)equation $\mathbf{a}.\mathbf{x} \boxtimes b$, we can partition encodings as follows.

- $[c_{\min}] = \{u \in (\Sigma_r^n)^+ \mid \mathbf{a}.\langle u\rangle_{r,n} < \min(b, -\|\mathbf{a}^+\|)\}$.
- $[c] = \{u \in (\Sigma_r^n)^+ \mid \mathbf{a}.\langle u\rangle_{r,n} = c\}$ for $c \in \mathbb{Z}$ with $\min(b, -\|\mathbf{a}^+\|) \leq c \leq \max(b, \|\mathbf{a}^-\|)$.
- $[c_{\max}] = \{u \in (\Sigma_r^n)^+ \mid \mathbf{a}.\langle u\rangle_{r,n} > \max(b, \|\mathbf{a}^-\|)\}$.

If two encodings u_1, u_2 are in the same class, then $\mathbf{a}.\langle u_1 v\rangle_{r,n} \boxtimes b \Leftrightarrow \mathbf{a}.\langle u_2 v\rangle_{r,n} \boxtimes b$ for all words $v \in (\Sigma_r^n)^*$.

Finally, the strongly connected components of NDDs will play a key role when generating exact formulas corresponding to represented sets. A first indication of the reasons why loops bring important information is provided in the following lemma, where one shows that for all scalar vectors $\mathbf{a} \in \mathbb{Z}^n$, for all encodings $u \in (\Sigma_r^n)^+$ and words $v \in (\Sigma_r^n)^*$, the sequence $\mathbf{a}.\langle u\rangle_{r,n}$, $\mathbf{a}.\langle uv\rangle_{r,n}$, $\mathbf{a}.\langle uv^2\rangle_{r,n}$, ... is either constant or strictly monotonic.

Lemma 84. *Let $\mathbf{a} \in \mathbb{Z}^n$ and let $u \in (\Sigma_r^n)^+$ be an encoding. For all words $v \in (\Sigma_r^n)^*$, there have three possibilities.*

- *If $\mathbf{a}.(\langle uv\rangle_{r,n} - \langle u\rangle_{r,n}) < 0$, then for all $k \in \mathbb{N}$, $\mathbf{a}.\langle uv^{k+1}\rangle_{r,n} < \mathbf{a}.\langle uv^k\rangle_{r,n}$.*
- *If $\mathbf{a}.(\langle uv\rangle_{r,n} - \langle u\rangle_{r,n}) = 0$, then for all $k \in \mathbb{N}$, $\mathbf{a}.\langle uv^{k+1}\rangle_{r,n} = \mathbf{a}.\langle uv^k\rangle_{r,n}$.*
- *If $\mathbf{a}.(\langle uv\rangle_{r,n} - \langle u\rangle_{r,n}) > 0$, then for all $k \in \mathbb{N}$, $\mathbf{a}.\langle uv^{k+1}\rangle_{r,n} > \mathbf{a}.\langle uv^k\rangle_{r,n}$.*

Proof. By definition of the encoding scheme, we have

$$\begin{aligned}\mathbf{a}.(\langle uv^{k+1}\rangle_{r,n} - \langle uv^k\rangle_{r,n}) &= \mathbf{a}.(r^{k\cdot|v|}\langle uv\rangle_{r,n} + \langle ov^k\rangle_{r,n} - r^{k\cdot|v|}\langle u\rangle_{r,n} - \langle ov^k\rangle_{r,n}) \\ &= r^{k\cdot|v|}\mathbf{a}.(\langle uv\rangle_{r,n} - \langle u\rangle_{r,n}).\end{aligned}$$

The claim is then immediate since $r^{k\cdot|v|} > 0$. □

Combining Lemmas 83 and 84, we deduce the following lemma.

Lemma 85. *For any inequation $\mathbf{a}.\mathbf{x} \leq b$, $\mathbf{a} \in \mathbb{Z}^n$, for all encodings $u \in (\Sigma_r^n)^+$ and words $v \in (\Sigma_r^n)^*$, there exists $k_{\min} \in N$ such that for all words $w \in (\Sigma_r^n)^*$, there are 3 possibilities.*

- *If $\mathbf{a}.\langle uv\rangle_{r,n} < \mathbf{a}.\langle u\rangle_{r,n}$, then $\mathbf{a}.\langle uv^k w\rangle_{r,n} \leq b$ for all $k \geq k_{\min}$.*
- *If $\mathbf{a}.\langle uv\rangle_{r,n} > \mathbf{a}.\langle u\rangle_{r,n}$, then $\mathbf{a}.\langle uv^k w\rangle_{r,n} > b$ for all $k \geq k_{\min}$.*
- *If $\mathbf{a}.\langle uv\rangle_{r,n} = \mathbf{a}.\langle u\rangle_{r,n}$, then $\mathbf{a}.\langle uv^k w\rangle_{r,n} = \mathbf{a}.\langle uv^{k+1} w\rangle_{r,n}$, for all $k \in \mathbb{N}$.*

Proof. Direct consequence of Lemmas 83 and 84. □

We now show that the previous lemma imposes that any minimal reduced NDD representing a set defined by a Boolean combination of inequations is permutation-free.

Lemma 86. *The reduced minimal NDD representing a set defined by a Boolean combination of finitely many linear inequations is permutation-free.*

Proof. Let $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be the reduced minimal NDD representing a set S defined by the formula $\bigvee_{i\in I}\bigwedge_{j\in J}\mathbf{a}_{ij}.\mathbf{x} \leq b_{ij}$, where I and J are finite subsets of $\mathbb{N}$ and $\mathbf{a}_{ij} \in \mathbb{Z}^n$ for all $i \in I$ and $j \in J$.

We prove by contradiction that $\mathcal{A}$ is permutation-free.

Suppose that $\mathcal{A}$ is not permutation-free. By definition, there exist a word $v \in (\Sigma_r^n)^*$ and a set $Q' = \{q_1, \ldots, q_m\} \subseteq Q$, with $m \geq 2$, such that $\hat{\delta}(q_i, v) = q_{i+1}$ for $1 \leq i \leq m-1$ and $\hat{\delta}(q_m, v) = q_1$.

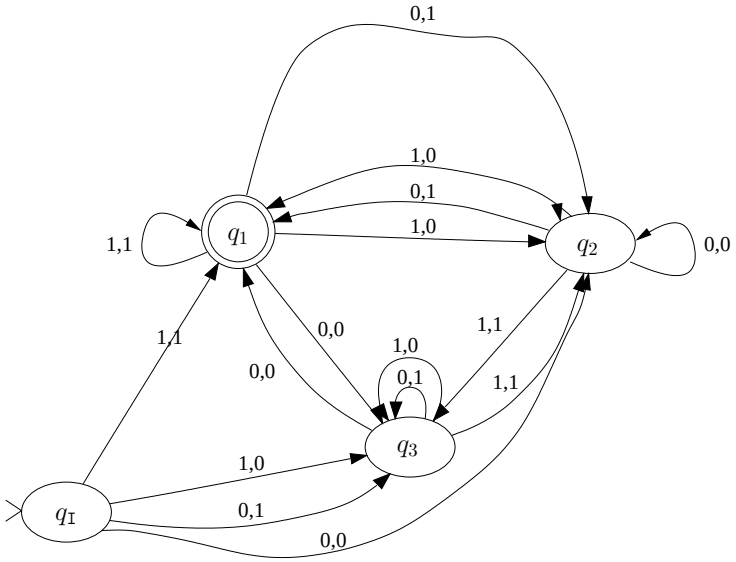


Figure 5.2: minimal NDD representing $S = \{(x, y) \in \mathbb{Z}^2 \mid x + y \equiv_3 1\}$

Since $\mathcal{A}$ is reduced minimal, without loss of generality, there exist an encoding $u \in (\Sigma_r^n)^+$ and a word $w \in (\Sigma_r^n)^*$ such that $\hat{\delta}(q_{\mathrm{I}}, u) = q_1$ and $uv^{lm}w \in L(\mathcal{A})$ but $uv^{lm+1}w \notin L(\mathcal{A})$ for all $l \in N$. Therefore, by definition, there exist $i' \in I$ and $j' \in J$ such that

$$\mathbf{a}_{i'j'}.\langle uv^{lm}w\rangle_{r,n} \leq b_{i'j'} \text{ and } \mathbf{a}_{i'j'}.\langle uv^{lm+1}w\rangle_{r,n} > b_{i'j'} \text{ for infinitely many } l. \tag{5.5}$$

Thanks to Lemma 85, there exists $k_{\min}$ such that either $\mathbf{a}_{i'j'}.\langle uv^{k}w\rangle_{r,n} \leq b_{i'j'}$ for all $k \geq k_{\min}$, $\mathbf{a}_{i'j'}.\langle uv^{k}w\rangle_{r,n} > b_{i'j'}$ for all $k \geq k_{\min}$, or $\mathbf{a}_{i'j'}.\langle uv^{k}w\rangle_{r,n} = \mathbf{a}_{i'j'}.\langle uv^{k+1}w\rangle_{r,n}$ for all $k \geq k_{\min}$. However, this contradicts (5.5). □

Remark 87. *The property of being permutation-free does not hold whenever congruence relations are introduced. For example, the reduced minimal NDD given in Fig.5.2 represents the set $S = \{(x, y) \in \mathbb{Z}^2 \mid x + y \equiv_3 1\}$ in basis 2, and is not permutation-free, since o makes a non-trivial permutation of the subset of states $\{q_1, q_3\}$.* □

5.4 Construction of NDDs from Quantifier-Free Formulas

In this section, we give a bound on the size of the minimal NDD representing a set defined by a quantifier-free Presburger formula as well as a procedure for constructing a deterministic NDD whose size is equal to the bound. The main result has been proved in [Kla04b, Kla04a] and is given here for the sake of completeness. Note that in [Kla04b, Kla04a], the construction uses extensively the concept of *pre-automaton* which is a DFA except that the set of accepting states is not specified. We give below a direct approach which is based on a right-invariant equivalence relation and does not involve pre-automata, but the idea as well as the complexity of the construction are essentially the same as in [Kla04b, Kla04a]. Note that the NDDs generated by the procedure given in this section are deterministic but in general not minimal. Efficient procedures generating minimal NDD for sets defined by a single equation and a single inequation are given in [WB00, Kla04b, Kla04a].

The basic idea for generating the NDD representing a set defined by a quantifier-free Presburger formula $\varphi(x_1, \ldots, x_n)$ is the following. Given a quantifier-free Presburger formula $\varphi(x_1, \ldots, x_n)$, one can compute finitely many formulas $\varphi_i(x_1, \ldots, x_n)$ satisfying the following requirements.

- Each vector in $\mathbb{Z}^n$ satisfies exactly one formula φ_i.
- Vectors satisfying the same formula have the same "possible future", i.e. if L_i is the set of encodings of vectors satisfying $\varphi_i(x_1, \ldots, x_n)$, for all $u_1, u_2 \in L_i$, for all words w, φ holds for $\langle u_1 w \rangle_{r,n}$ iff φ holds for $\langle u_2 w \rangle_{r,n}$.
- φ is equivalent to a disjunction of some formulas φ_i.

Then one defines a right-invariant equivalence relation R_φ on words in $(\Sigma_r^n)^*$ such that the equivalence classes of R_φ correspond to sets of encodings satisfying the individual formulas φ_i. The construction of the NDD consists in associating one state to each equivalence class and defining the transition function based on the equivalence classes of R_φ.

Let $\varphi(x_1, \ldots, x_n)$ be a quantifier-free Presburger formula, i.e. $\varphi(x_1, \ldots, x_n)$ is a Boolean combination of atomic formulas of the form $\mathbf{a}.\mathbf{x} \boxtimes b$, with $\mathbf{a} \in \mathbb{Z}^n$ and $\boxtimes \in \{<, \leq, =, \geq, >\}$, or $\mathbf{a}.\mathbf{x} \equiv_m b$, with $\mathbf{a} \in Z^n$ and $m \in \mathbb{Z}$, $m \geq 1$. In the sequel, the symbol $n_\boxtimes$ (resp. $n_\equiv$) denotes the number of different vectors $\mathbf{a}$ such

that $\mathbf{a}.\mathbf{x}\boxtimes b$ (resp. $\mathbf{a}.\mathbf{x} \equiv_m b$) appears in φ for some $b \in \mathbb{Z}$ (and $m \in \mathbb{Z}$). Similarly, the symbol n_m denotes the number of integer m such that $\mathbf{a}.\mathbf{x} \equiv_m b$ appears in φ for some $\mathbf{a} \in \mathbb{Z}^n$ and $m \in \mathbb{Z}$.

Let $c_{\min}, c_{\max}, m_{\max} \in \mathbb{Z}$, $\mathbf{a}_{\boxtimes,1}, \ldots, \mathbf{a}_{\boxtimes,n_\boxtimes}$, $\mathbf{a}_{\equiv,1}, \ldots, \mathbf{a}_{\equiv,n_\equiv} \in \mathbb{Z}^n$ and $m_1, \ldots, m_{n_m} \in \mathbb{N}$ such that

- for all atomic formulas in φ of the form $\mathbf{a}.\mathbf{x} \boxtimes b$, $\mathbf{a} = \mathbf{a}_{\boxtimes,i}$ for some $i \in \{1, \ldots, n_\boxtimes\}$, $c_{\min} < \min(-\| \mathbf{a}^+\|, b)$, and $c_{\max} > \max(\| \mathbf{a}^-\|, b)$,
- for all atomic formulas in φ of the form $\mathbf{a}.\mathbf{x} \equiv_m b$, $\mathbf{a} = \mathbf{a}_{\equiv,i}$ for some $i \in \{1, \ldots, n_\equiv\}$, $m = m_j$ for some $j \in \{1, \ldots, n_m\}$, and $m \leq m_{\max}$.

For all $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ and $\mathbf{m} \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$, we define $\varphi_{\mathbf{c},\mathbf{m}}$ as follows.

$$\varphi_{\mathbf{c},\mathbf{m}}(\mathbf{x}) =_{def} \bigwedge_{i \in \{1,\ldots,n_\boxtimes\}} \mathbf{a}_{\boxtimes,i}.\mathbf{x} \boxtimes_i \mathbf{c}[i] \wedge \bigwedge_{\substack{i \in \{1,\ldots,n_\equiv\},\\ j \in \{1,\ldots,n_m\}}} \mathbf{a}_{\equiv,i}.\mathbf{x} \equiv_{m_j} \mathbf{m}[i+(j-1)\cdot n_\equiv],$$

where $\boxtimes_i$ is $\begin{cases} \leq \text{ if } \mathbf{c}[i] = c_{\min} \\ = \text{ if } c_{\min} < \mathbf{c}[i] < c_{\max} \\ \geq \text{ if } \mathbf{c}[i] = c_{\max} \end{cases}$.

Example 88. *Let* $\phi(x, y) =_{def} (x + y = 1 \vee x + y \leq -3 \vee 2x - y \leq -4) \wedge (x + 2y \equiv_3 1 \vee x + 2y \equiv_4 0 \vee 3x + y \equiv_5 2)$.
By definition, one can choose

- $n_\boxtimes = 2$ *with* $\mathbf{a}_{\boxtimes,1} = (1, 1)$ *and* $\mathbf{a}_{\boxtimes,2} = (2, -1)$,
- $n_\equiv = 2$ *with* $\mathbf{a}_{\equiv,1} = (1, 2)$ *and* $\mathbf{a}_{\equiv,2} = (3, 1)$,
- $n_m = 3$ *with* $m_1 = 3$, $m_2 = 4$ *and* $m_3 = 5$,
- $c_{\min} = -5$, $c_{\max} = 2$, *and* $m_{\max} = 5$.

Let $\mathbf{c} \in \{-5, \ldots, 2\}^2$ *and* $\mathbf{m} \in \{1, \ldots, 3\}^2 \times \{1, \ldots, 4\}^2 \times \{1, \ldots, 5\}^2$. *By definition, we have*

$$\begin{aligned} \varphi_{\mathbf{c},\mathbf{m}}(x, y) \;=_{def}\; & x + y \boxtimes_1 \mathbf{c}[1] \wedge 2x - y \boxtimes_2 \mathbf{c}[2] \\ & \wedge x + 2y \equiv_3 \mathbf{m}[1] \wedge 3x + y \equiv_3 \mathbf{m}[2] \wedge x + 2y \equiv_4 \mathbf{m}[3] \\ & \wedge 3x + y \equiv_4 \mathbf{m}[4] \wedge x + 2y \equiv_5 \mathbf{m}[5] \wedge 3x + y \equiv_5 \mathbf{m}[6], \end{aligned}$$

where $\boxtimes_i$ *is* $\begin{cases} \leq \textit{ if } \mathbf{c}[i] = -5 \\ = \textit{ if } -5 < \mathbf{c}[i] < 2 \\ \geq \textit{ if } \mathbf{c}[i] = 2 \end{cases}$. □

We have the following lemmas.

Lemma 89. *For each* $\mathbf{x} \in \mathbb{Z}^n$*, there is exactly one pair* $(\mathbf{c}, \mathbf{m})$ *with* $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_{\boxtimes}}$ *and* $\mathbf{m}\{1, \ldots, m_1\}^{n_{\equiv}} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_{\equiv}}$*, such that* $\varphi_{\mathbf{c},\mathbf{m}}$ *holds for* $\mathbf{x}$*.*

Proof. This is a direct consequence of the definition of the formulas $\varphi_{\mathbf{c},\mathbf{m}}$. □

Lemma 90. *Each formula* $\varphi_{\mathbf{c},\mathbf{m}}$*, with* $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_{\boxtimes}}$ *and* $\mathbf{m} \in \{1, \ldots, m_1\}^{n_{\equiv}} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_{\equiv}}$*, satisfies the following property.*

For all encodings $u_1, u_2 \in (\Sigma_r^n)^+$ *and words* $v \in (\Sigma_r^n)^*$*, if* $\varphi_{\mathbf{c},\mathbf{m}}$ *holds for both* u_1 *and* u_2*, then* φ *holds for* $\langle u_1 v \rangle_{r,n}$ *iff* φ *holds for* $\langle u_2 v \rangle_{r,n}$*.*

Proof. Suppose that $\varphi_{\mathbf{c},\mathbf{m}}$ holds for u_1 and u_2. By construction, for all $\mathbf{a}.\mathbf{x} \boxtimes b$ appearing in φ, either $\mathbf{a}.\langle u_1 \rangle_{r,n} \leq c_{\min} \wedge \mathbf{a}.\langle u_2 \rangle_{r,n} \leq c_{\min}$, $\mathbf{a}.\langle u_1 \rangle_{r,n} = \mathbf{a}.\langle u_2 \rangle_{r,n}$, or, $\mathbf{a}.\langle u_1 \rangle_{r,n} \geq c_{\max} \wedge \mathbf{a}.\langle u_2 \rangle_{r,n} \geq c_{\max}$. Therefore, by definition of $c_{\min}$ and $c_{\max}$, and thanks to Lemma 83, we have either $\mathbf{a}.\langle u_1 v \rangle_{r,n} \leq c_{\min} < b \wedge \mathbf{a}.\langle u_2 v \rangle_{r,n} \leq c_{\min} < b$, $\mathbf{a}.\langle u_1 v \rangle_{r,n} = \mathbf{a}.\langle u_2 v \rangle_{r,n}$, or, $\mathbf{a}.\langle u_1 v \rangle_{r,n} \geq c_{\max} > b \wedge \mathbf{a}.\langle u_2 v \rangle_{r,n} \geq c_{\max} > b$. So, the formula $\mathbf{a}.\mathbf{x} \boxtimes b$ holds for $\langle u_1 v \rangle_{r,n}$ iff it holds for $\langle u_2 v \rangle_{r,n}$.

In addition, for all $\mathbf{a}.\mathbf{x} \equiv_m b$ appearing in φ, by construction, there exist $m = m_j$ for some $j \in \{1, \ldots, n_m\}$ and $\mathbf{a} = \mathbf{a}_i$ for some $i \in \{1, \ldots, n_{\equiv}\}$. So, by hypothesis, $\mathbf{a}.\langle u_1 \rangle_{r,n} \equiv_m \mathbf{a}.\langle u_2 \rangle_{r,n}$, and thanks to Lemma 81, $\mathbf{a}.\langle u_1 v \rangle_{r,n} \equiv_m \mathbf{a}.\langle u_2 v \rangle_{r,n}$. Therefore, the formula $\mathbf{a}.\mathbf{x} \equiv_m b$ holds for $\langle u_1 v \rangle_{r,n}$ iff it holds for $\langle u_2 v \rangle_{r,n}$.

We conclude that φ holds for $\langle u_1 v \rangle_{r,n}$ iff it holds for $\langle u_2 v \rangle_{r,n}$. □

Lemma 91. *There exists a sequence of pairs* $(\mathbf{c}_1, \mathbf{m}_1)$*, …,* $(\mathbf{c}_p, \mathbf{m}_p)$ *with* $\mathbf{c}_i \in \{c_{\min}, \ldots, c_{\max}\}^{n_{\boxtimes}}$ *and* $\mathbf{m}_i \in \{1, \ldots, m_1\}^{n_{\equiv}} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_{\equiv}}$ *for all* $i \in \{1, \ldots, p\}$ *such that for all* $\mathbf{x} \in \mathbb{Z}^n$*, we have*

$$\varphi(\mathbf{x}) \Leftrightarrow \bigvee_{i \in \{1,\ldots,p\}} \varphi_{\mathbf{c}_i,\mathbf{m}_i}(\mathbf{x}).$$

Proof. Thanks to Lemma 89, we can partition the encodings $u \in (\Sigma_r^n)^+$ based on the formula $\varphi_{\mathbf{c},\mathbf{m}}$ satisfied by $\langle u \rangle_{r,n}$. Thanks to Lemma 90, if u_1 and u_2 are in the same partition, φ holds for $\langle u_1 \rangle_{r,n}$ iff it holds for $\langle u_2 \rangle_{r,n}$. Let $(\mathbf{c}_1, \mathbf{m}_1)$, …, $(\mathbf{c}_p, \mathbf{m}_p)$ be the pairs such that φ and $\varphi_{\mathbf{c}_i,\mathbf{m}_i}$ holds for some $\mathbf{x}$. We conclude that for all $\mathbf{x} \in \mathbb{Z}^n$, $\varphi(\mathbf{x}) \Leftrightarrow \bigvee_{i \in \{1,\ldots,p\}} \varphi_{\mathbf{c}_i,\mathbf{m}_i}(\mathbf{x})$. □

We define the relation R_φ as the smallest binary relation on words in $(\Sigma_r^n)^*$ satisfying the following requirements.

- $(\varepsilon, \varepsilon) \in R_\varphi$,
- $(u, v) \in R_\varphi$ if u, v are valid encodings (i.e. $|u|, |v| \geq 1$ and their first symbol is a sign symbol) and for all $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ and $\mathbf{m} \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$, $\varphi_{\mathbf{c},\mathbf{m}}(\langle u \rangle_{r,n}) \Leftrightarrow \varphi_{\mathbf{c},\mathbf{m}}(\langle v \rangle_{r,n})$.

Lemma 92. *The binary relation R_φ is an equivalence relation and its equivalence classes are*

- $\{\varepsilon\}$,
- $(\Sigma_r^n)^+ \setminus \{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0, r-1\}^n\}$,
- *the nonempty sets* $\{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0, r-1\}^n \wedge \varphi_{\mathbf{c},\mathbf{m}}(\langle \alpha u \rangle_{r,n})\}$ *for all* $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ *and* $\mathbf{m} \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$.

Proof. By definition, R_φ is reflexive, symmetric and transitive, and therefore, R_φ is an equivalence relation. In addition, for all valid encodings $u, v \in (\Sigma_r^n)^+$, if $\varphi_{\mathbf{c},\mathbf{m}}$ holds for both $\langle u \rangle_{r,n}$ and $\langle v \rangle_{r,n}$, then $(u, v) \in R_\varphi$, i.e. u and v are in the same equivalence class. Finally, since there is exactly one formula $\varphi_{\mathbf{c},\mathbf{m}}$ holding for each integer vector in $\mathbb{Z}^n$, the sets $\{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0, r-1\}^n \wedge \varphi_{\mathbf{c},\mathbf{m}}(\langle \alpha u \rangle_{r,n})\}$, $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ and $\mathbf{m} \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$, are disjoint. □

In order to construct the NDD representing the integer vectors satisfying φ in a way similar to the automaton construction of Lemma 49, one needs to show that R_φ is right-invariant with respect to concatenation, i.e. for all $u, v, w \in (\Sigma_r^n)^*$, if $(u, v) \in R_\varphi$, then $(uw, vw) \in R_\varphi$. This is a direct consequence of the following lemma.

Lemma 93. *Let $u, v \in (\Sigma_r^n)^+$ be valid encodings and let $\mathbf{c} \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ and $\mathbf{m} \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$.*

If $\varphi_{\mathbf{c},\mathbf{m}}$ holds for both $\langle u \rangle_{r,n}$ and $\langle v \rangle_{r,n}$, then $\varphi_{\mathbf{c}',\mathbf{m}'}(\langle uw \rangle_{r,n}) \Leftrightarrow \varphi_{\mathbf{c}',\mathbf{m}'}(\langle vw \rangle_{r,n})$ for all $w \in (\Sigma_r^n)^*$, $\mathbf{c}' \in \{c_{\min}, \ldots, c_{\max}\}^{n_\boxtimes}$ and $\mathbf{m}' \in \{1, \ldots, m_1\}^{n_\equiv} \times \cdots \times \{1, \ldots, m_{n_m}\}^{n_\equiv}$.

Proof. Suppose that $\varphi_{\mathbf{c},\mathbf{m}}$ holds for both $\langle u \rangle_{r,n}$ and $\langle v \rangle_{r,n}$.

By construction, for all $i \in \{1, \ldots, n_\boxtimes\}$, either $\mathbf{a}_i.\langle u \rangle_{r,n} \leq c_{\min} \wedge \mathbf{a}_i.\langle v \rangle_{r,n} \leq c_{\min}$, $\mathbf{a}_i.\langle u \rangle_{r,n} = \mathbf{a}_i.\langle v \rangle_{r,n}$, or, $\mathbf{a}_i.\langle u \rangle_{r,n} \geq c_{\max} \wedge \mathbf{a}_i.\langle v \rangle_{r,n} \geq c_{\max}$. Therefore, by definition of $c_{\min}$ and $c_{\max}$, and thanks to Lemmas 81 and 83, we have either

$\mathbf{a}_i.\langle uw\rangle_{r,n} \leq c_{\min} \wedge \mathbf{a}_i.\langle vw\rangle_{r,n} \leq c_{\min}$, $\mathbf{a}_i.\langle uw\rangle_{r,n} = \mathbf{a}_i.\langle vw\rangle_{r,n}$, or, $\mathbf{a}_i.\langle uw\rangle_{r,n} \geq c_{\max} \wedge \mathbf{a}_i.\langle vw\rangle_{r,n} \geq c_{\max}$.

In addition, for all $i \in \{1,\ldots,n_{\equiv}\}$ and $j \in \{1,\ldots,n_m\}$, $\mathbf{a}_i.\langle u\rangle_{r,n} \equiv_{m_j} \mathbf{a}_i.\langle v\rangle_{r,n}$, and thanks to Lemma 81, $\mathbf{a}_i.\langle uw\rangle_{r,n} \equiv_{m_j} \mathbf{a}_i.\langle vw\rangle_{r,n}$.

The claim is then a consequence of the definition of the formulas $\varphi_{\mathbf{c},\mathbf{m}}$. □

Lemma 94. *The binary relation R_φ is right-invariant.*

Proof. Let $u, v \in (\Sigma_r^n)^*$ and suppose that $(u,v) \in R_\varphi$. By definition, either $u = v = \varepsilon$ or u and v are valid encodings and $\varphi_{\mathbf{c},\mathbf{m}}$ holds for both $\langle u\rangle_{r,n}$ and $\langle v\rangle_{r,n}$, for some $\mathbf{c} \in \{c_{\min},\ldots,c_{\max}\}$. In both cases, thanks to Lemma 93 and by definition of R_φ, for all words $w \in (\Sigma_r^n)^*$, $(uw, vw) \in R_\varphi$. □

Theorem 95. *There exists a function* NDD_CONSTRUCT*, which, given the quantifier-free Presburger formula $\varphi(x_1,\ldots,x_n)$, returns a deterministic NDD $\mathcal{A}$ in strong normal form accepting the set of integer vectors for which φ holds.*

The NDD $\mathcal{A}$ has at most $(|c_{\min}| + |c_{\max}| + 1)^{n_{\boxtimes}} \cdot m_{\max}^{n_{\equiv}\cdot n_m}$ states and the time complexity of NDD_CONSTRUCT *is $\mathcal{O}(|\Sigma_r^n| \cdot (|c_{\min}| + |c_{\max}| + 1)^{n_{\boxtimes}} \cdot m_{\max}^{n_{\equiv}\cdot n_m})$.*

Proof. Thanks to Lemmas 92 and 94, R_φ is a right-invariant equivalence relation of finite index, whose equivalence classes are

- $\{\varepsilon\}$,
- $\Sigma_r^n)^+ \setminus \{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0, r-1\}^n\}$, and
- the nonempty sets $\{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0,r-1\}^n \wedge \varphi_{\mathbf{c},\mathbf{m}}(\langle \alpha u\rangle_{r,n})\}$ for all $\mathbf{c} \in \{c_{\min},\ldots,c_{\max}\}^{n_{\boxtimes}}$ and $\mathbf{m} \in \{1,\ldots,m_1\}^{n_{\equiv}} \times \cdots \times \{1,\ldots,m_{n_m}\}^{n_{\equiv}}$.

Also, thanks to Lemma 91, the set of encodings of vectors for which φ holds is a union of equivalence classes. Therefore, thanks to Lemma 49, one can construct the NDD representing the integer vectors satisfying φ. More precisely, if $[u]_{R_\varphi}$ denotes the equivalence class of R_φ containing the word u, $\mathcal{A}$ is $(Q, \Sigma, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ where

- $Q = \{[u]_{R_\varphi} \mid u \in \Sigma^*\}$.
- For each $\alpha \in \Sigma_r^n$ and $[u]_{R_\varphi} \in Q$, $\delta([u]_{R_\varphi}, \alpha) = [u\alpha]_{R_\varphi}$.
- $q_{\mathtt{I}} = [\varepsilon]_{R_\varphi}$.
- $Q_{\mathtt{F}} = \{[u]_{R_\varphi} \mid u \in \{0,r-1\}^n(\Sigma_r^n)^* \wedge \varphi(\langle u\rangle_{r,n})\}$.

Note that the construction can be done incrementally, starting with the initial state $[\varepsilon]$, and given a state $[u]_{R_\varphi} \in Q$, for each $\alpha \in \Sigma_r^n$, one computes $[u\alpha]_{R_\varphi} \in Q$ as follows. If $u\alpha$ is not a valid encoding, then $[u\alpha]_{R_\varphi}$ is $(\Sigma_r^n)^+ \setminus \{\alpha u \in (\Sigma_r^n)^+ \mid \alpha \in \{0, r-1\}^n\}$. Otherwise, $[u\alpha]_{R_\varphi}$ is the equivalence class associated to the formula $\varphi_{\mathbf{c},\mathbf{m}}$ such that for all $i \in \{1, \ldots, n_\boxtimes\}$,

$$\begin{array}{ll} \mathbf{c}[i] = c_{\min} & \text{if } \mathbf{a}_i.\langle u\alpha\rangle_{r,n} \leq c_{\min} \\ \mathbf{c}[i] = \mathbf{a}_i.\langle u\alpha\rangle_{r,n} & \text{if } c_{\min} < \mathbf{a}_i.\langle u\alpha\rangle_{r,n} < c_{\max} \\ \mathbf{c}[i] = c_{\max} & \text{if } \mathbf{a}_i.\langle u\alpha\rangle_{r,n} \geq c_{\max} \end{array}$$

and for all $i \in \{1, \ldots, n_\equiv\}$, $j \in \{1, \ldots, n_m\}$,

$$\mathbf{m}[i + j \cdot n_\equiv] = \mathbf{a}_i.\langle u\alpha\rangle_{r,n} \mod m_j.$$

By construction, $\mathcal{A}$ has at most $(|c_{\min}| + |c_{\max}| + 1)^{n_\boxtimes} \cdot m_{\max}^{n_\equiv \cdot n_m}$ states and since each state is handled only once, the time complexity of the algorithm is $\mathcal{O}(|\Sigma_r^n| \cdot (|c_{\min}| + |c_{\max}| + 1)^{n_\boxtimes} \cdot m_{\max}^{n_\equiv \cdot n_m})$. □

5.5 Other Encoding Schemes

The synchronous encoding scheme is not the only scheme that is suited for integer vectors. In this section, we present two encoding schemes closely related to the synchronous encoding scheme, the *reverse synchronous encoding scheme* and the *synchronous interleaved encoding scheme* [Boi99].

5.5.1 Reverse Synchronous Encoding Scheme

The *Reverse Synchronous Encoding Scheme* follows the same rules as the *Synchronous Encoding Scheme* except that the digits are read from the least significant digit one to the most significant one rather than the other way round.

Definition 96. *Given an* encoding basis $r > 1$, *a word* $w = a_0 \ldots a_p$ *with* $p \in \mathbb{N}$, $a_p \in \{0, r-1\}$ *and* $a_i = \{0, \ldots, r-1\}$ *for* $i \in \{0, \ldots, p-1\}$, *is an* lsdf r-encoding of an integer $z \in \mathbb{Z}$, *denoted by* $\langle w\rangle_r^{\mathrm{R}} = z$, *if* $z = -r^p \cdot \frac{a_p}{r-1} + \sum_{i=0}^{p-1} a_i r^i$. □

Note that given an integer z, and $0 \leq a_i < r$, $i \in \{0, \ldots, p\}$, $\langle a_0 \ldots a_p\rangle_r^{\mathrm{R}} = z$ if and only if $\langle a_p \ldots a_0\rangle_r = z$.

Definition 97. *Given a dimension* $n \geq 0$ *and an encoding basis* $r > 1$, *a word* $w = \alpha_0 \ldots \alpha_p$ *in* $(\Sigma_r^n)^*$, *with* $p \in \mathbb{N}$ *and* $p \geq 1$, *is an* lsdf r-encoding of an integer

vector $\mathbf{z} \in \mathbb{Z}^n$, *denoted* $\langle w \rangle^{\mathrm{R}}_{r,n} = \mathbf{z}$ *if for each* $j \in \{1, \ldots, n\}$, $\langle \alpha_0[j] \ldots \alpha_p[j] \rangle^{\mathrm{R}}_r = \mathbf{z}[j]$.

The reverse synchronous encoding scheme $E_{R(r)}$ *is the relation that associates to a vector* $\mathbf{z} \in \mathbb{Z}^n$ *the words* $w \in (\Sigma_r^n)^*$ *such that* $\mathbf{z} = \langle w \rangle^{\mathrm{R}}_{r,n}$. □

Given a set $S \subseteq \mathbb{Z}^n$, if the language $L^{\mathrm{R}}(S)$ containing all the lsdf r-encodings of all the vectors in S is regular, then S is *recognizable* with respect to the reverse synchronous encoding scheme $E_{R(r)}$.

Theorem 98. *A set* $S \subseteq \mathbb{Z}^n$ *is recognizable with respect to the synchronous encoding scheme* $E_{S(r)}$ *if and only if* S *is recognizable with respect to the reverse synchronous scheme.*

Proof. This is a direct consequence of the facts that w is an msdf r-encoding of a vector $\mathbf{z} \in \mathbb{Z}^n$ if and only if w^{R} is a lsdf r-encoding of $\mathbf{z}$ and that, thanks to Propositions 47 and 64, a language L is accepted by a DFA if and only if L^{R} is accepted by a DFA, where $L^{\mathrm{R}} = \{w^{\mathrm{R}} \mid w \in L\}$. □

Although the class of sets that are recognizable with respect to the synchronous encoding scheme $E_{S(r)}$ and the class of sets recognizable with respect to the reverse encoding scheme $E_{R(r)}$ are identical, this does not mean that there are equivalent in practice. Indeed, the conversion from one representation to the other implies a determinization and it is known [MF71, Moo71] that there exist languages L such that the (reduced) minimal automaton accepting L is exponentially larger that some FA accepting L. So, since Theorem 76 relating the size of the complete minimal DFA representing a set defined by a Presburger formula φ with respect to the synchronous encoding scheme $E_{S(r)}$ does not apply when considering the reverse synchronous scheme, the best available bound on the size of the complete minimal DFA accepting the set defined by φ with respect to the reverse synchronous encoding scheme is $2^{2^{2^{2^{|\varphi|}}}}$, although the current conjecture is that the worst case complexity are identical for both encoding schemes. Note also that the worst case presented in [Kla04b, Kla04a] is very peculiar and we do not expect such a complexity for most practical cases, as confirmed by experiments.

In order to provide some ideas on the differences in practice between the sizes of the automata when using the different encoding schemes, we compare the sizes of NDDs using both the synchronous encoding scheme and the reverse encoding scheme in five types of simple formulas : linear equations, linear inequations, congruence relations with modulos relatively prime to the encoding basis, congruence relations with modulos powers of the encoding basis, and formulas of type

$b_{\min} \leq \mathbf{a}.\mathbf{x} \leq b_{\max}$, which we call *intervals*. In addition, we analyze the impact of the sign symbol when using both encoding.

In all formulas, we used 8 variables, i.e. the represented sets are subsets of $\mathbb{Z}^8$, and all scalars used in the formulas were integer numbers randomly chosen within some bounds. We use 2 as the encoding basis.

In Figures[2] 5.3,5.4, 5.5,5.6 and 5.7, the sets considered are all subsets of $\mathbb{N}^8$. Then in Figures 5.8,5.9, 5.10,5.11 and 5.12, we dropped the requirement on the sign of the solutions. In each graph, the sizes of the automata obtained when using the synchronous encoding scheme are plotted on the abscissa whereas the sizes of the automata obtained when using the reverse synchronous encoding are plotted on the ordinate.

According to Figures 5.3,5.4, 5.5,5.6 and 5.7, the sizes of the automata accepting encodings of positive solutions of equations, inequations or congruences modulo p where p is prime relatively to the encoding basis does not depend on the choice of encoding scheme. This is no longer true in the case of congruences modulo a power of the basis, where the size is almost doubled when using the reverse encoding scheme. The difference is even more striking with intervals.

When considering the sign symbols, it appears that using the reverse synchronous encoding scheme incurs an additional cost. We explain this as follows. With the reverse synchronous encoding scheme, since the semantic of a symbol is different whether the symbol is the last symbol of an encoding, i.e. the sign symbol, or not, one has to distinguish final states from the others, and the extra information regarding the accepting status leads to splitting some states. For example, when representing the equation $2x_1 + 3x_2 = 0$ with the reverse encoding scheme, the prefixes $u_1 = (1,0)(0,1)(1,0)$ and $u_2 = (0,0)(0,0)(1,1)$ are equivalent when at least one symbol must be read. Indeed, in both cases, 3 symbols have been read and since $\langle u_1 o\rangle^{\mathrm{R}}_{r,n} = (5,2)$ and $\langle u_2 o\rangle^{\mathrm{R}}_{r,n} = (4,4)$, the value of the left-hand side of $2x_1 + 3x_2 = 0$ is 16 when substituting (x_1, x_2) by $\langle u_1 o\rangle^{\mathrm{R}}_{r,n}$ or $\langle u_2 o\rangle^{\mathrm{R}}_{r,n}$. However, if the third symbol is the sign symbol, then $\langle u_1\rangle^{\mathrm{R}}_{r,n} = (-3,2)$ and $\langle u_2\rangle^{\mathrm{R}}_{r,n} = (-4,-4)$, and in this case, the value of the left-hand side is 0 for u_1 and -20 for u_2, and the encodings are no longer equivalent. With the synchronous encoding scheme, the above phenomenon does not occur since the sign symbols label transitions rooted at the initial state and they do no longer appear afterward, i.e. they do not label transitions rooted at some other state.

Based on the experimental results presented above, we conclude that in the

[2]We slightly modified the encoding schemes by removing the sign symbol, (which would have been o for all encodings).

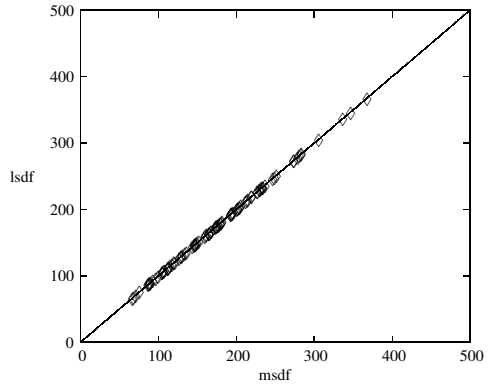


Figure 5.3: msdf vs lsdf in equations (no sign)

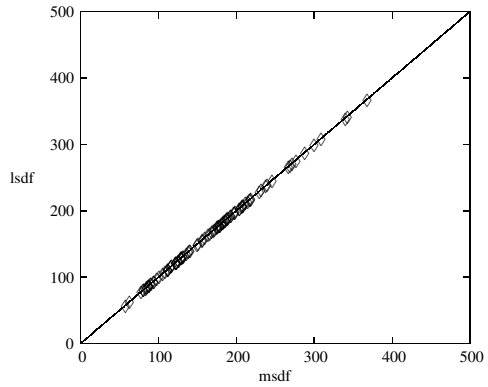


Figure 5.4: msdf vs lsdf in inequations (no sign)

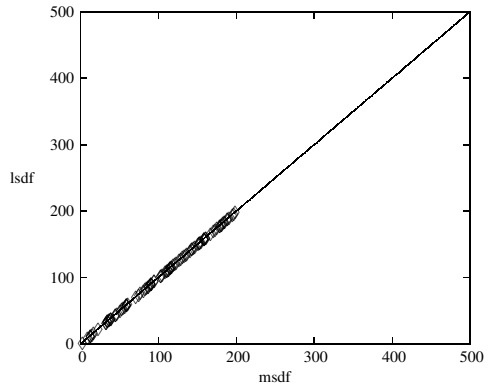


Figure 5.5: msdf vs lsdf in congruences with modulo prime to basis (no sign)

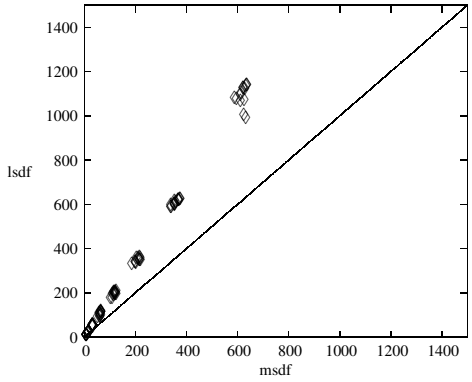


Figure 5.6: msdf vs lsdf in congruences with modulo power of the basis (no sign)

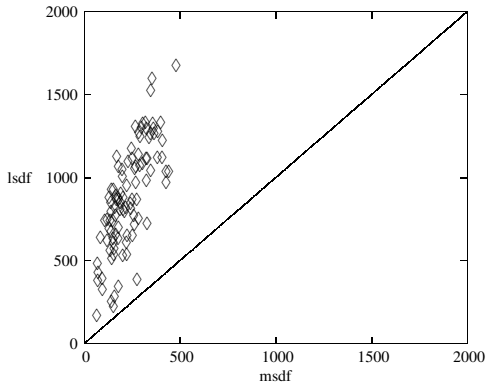


Figure 5.7: msdf vs lsdf in intervals (no sign)

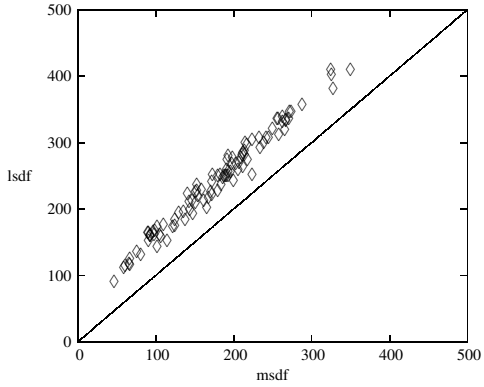


Figure 5.8: msdf vs lsdf in equations

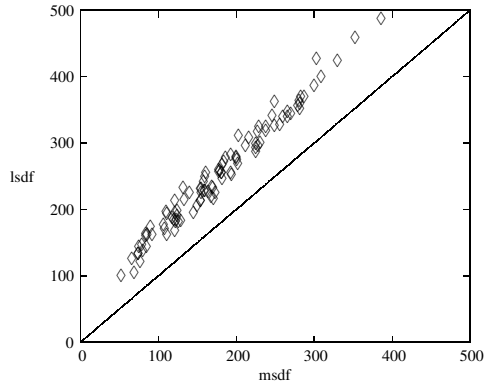


Figure 5.9: msdf vs lsdf in inequations

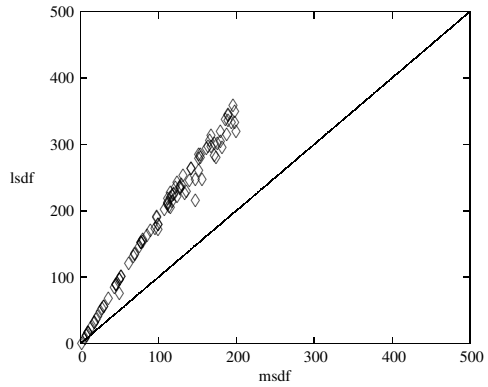


Figure 5.10: msdf vs lsdf in congruences with modulo prime to basis

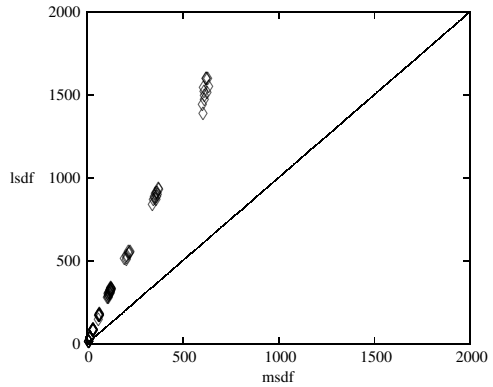


Figure 5.11: msdf vs lsdf in congruences with modulo power of the basis

case of simple sets, the synchronous encoding scheme leads to NDDs with fewer states than those produced when using the reverse encoding scheme. We expect that this observation also holds for more general sets on average.

5.5.2 Synchronous Interleaved Encoding Scheme

As mentioned in Section 5.1, when using the synchronous encoding scheme, each symbol α_i of an r-encoding $\alpha_p \dots \alpha_0$ of a vector $\mathbf{z} \in \mathbb{Z}^n$ is basically a vector of digits, and $\alpha_i[j]$ is the ith digit of an r-encoding of $\mathbf{z}[j]$. In the *synchronous interleaved encoding scheme*, also referred to as the *serial encoding*, the digits of the encodings of the components are read sequentially rather than simultaneously. Formally, we have the following definitions.

Definition 99. *Given a dimension $n \geq 0$ and an encoding basis $r > 1$, a word $w = a_{p,1} \dots a_{p,n} \dots a_{0,n}$ in $(\Sigma_r)^*$, with $p \in \mathbb{N}$ and $p \geq 1$, is a* (msdf) serialized r-encoding of an integer vector $\mathbf{z} \in \mathbb{Z}^n$, *denoted $\langle w \rangle^{\mathrm{I}}_{r,n} = \mathbf{z}$, if for each $j \in \{1, \dots, n\}$, $\langle \alpha_{p,j} \alpha_{p-1,j} \dots \alpha_{0,j} \rangle_r = \mathbf{z}[j]$.*

The synchronous interleaved encoding scheme *(also called* serial encoding scheme*), $E_{I(r)}$ is the relation that associates to a vector $\mathbf{z} \in \mathbb{Z}^n$ the words $w \in (\Sigma_r)^*$ such that $\mathbf{z} = \langle w \rangle^{\mathrm{I}}_{r,n}$.* □

The sets of vectors that are recognizable with respect to synchronous interleaved encoding scheme are exactly the ones that are recognizable with respect to the synchronous encoding scheme.

Theorem 100. *A set $S \subseteq \mathbb{Z}^n$ is recognizable with respect to the synchronous encoding scheme $E_{S(r)}$ if and only if S is recognizable with respect to the synchronous interleaved encoding scheme.*

Proof. A DFA $\mathcal{A}' = (Q', \Sigma'_r, \delta', q'_{\mathtt{I}}, Q'_{\mathtt{F}})$ representing a set $S \subseteq \mathbb{Z}^n$ with respect to the synchronous interleaved encoding scheme $E_{I(r)}$ is easily generated from a DFA $\mathcal{A} = (Q, \Sigma_r, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ representing S with respect to the synchronous encoding scheme by adding intermediate states. That is, if $\delta(q, \alpha) = q'$, one adds $n - 1$ states $q_1, \dots, q_{n-1} \in Q'$ such that $\delta'(q, \alpha[1]) = q_1$, $\delta'(q_{i-1}, \alpha[i]) = q_i$, $i \in \{2, \dots, n-2\}$ and $\delta'(q_{n-1}, \alpha[n]) = q'$.

The reciprocal transformations can be achieved by removing the intermediate states. □

The interest of the synchronous interleaved encoding scheme is that some redundant information in the transition function can now be gotten rid of by sharing intermediate states.

This is shown in Figures 5.13 and 5.14, which display the reduced minimal DFA representing $\mathbb{Z}^3$ with respect to the synchronous encoding scheme and synchronous interleaved scheme respectively.

In order to analyze the impact of using the synchronous interleaved scheme, we do a comparison of sizes of NDDs when using the synchronous encoding scheme $E_{S(r)}$ and the synchronous interleaved one $E_{I(r)}$, similar to what has been presented in Section 5.5.1, i.e. we consider sets corresponding to linear equations, linear inequations, congruence relations with modulos relatively prime to the encoding basis, congruence relations with modulos powers of the encoding basis, and intervals.

In all formulas, we used 8 variables, i.e. the represented sets are subsets of $\mathbb{Z}^8$, and all scalars used in the formulas were integer numbers randomly chosen within some bounds. We use 2 as the encoding basis.

The results are presented in Figures 5.15, 5.16, 5.17, 5.18 and 5.19.

From the figures, we see that, as a rule of thumb, the number of states of the complete minimal DFA representing a set $S \subseteq \mathbb{Z}^n$ is multiplied by n when using $E_{I(r)}$ as compared to $E_{S(r)}$. In order to get the number of transitions in the complete minimal DFA, one has to multiply the number of states by the number of symbols in the alphabet, that is, r^n and r when using $E_{S(r)}$ and $E_{I(r)}$ respectively. So, from the figures, we conclude that the number of transitions is exponentially smaller in general when using the synchronous interleaved scheme. This suggests that in general, the transition relations of NDDs using the synchronous encoding scheme contain a lot of redundancy which can be gotten rid of by using the synchronous interleaved scheme.

In the sequel, all results will be stated when using the synchronous encoding scheme $E_{S(r)}$ since it presents the advantage that the label of any path rooted at the initial state is an encoding, and so, it facilitates the interpretation of the results. However, in order to test the algorithms presented in the sequel, we will adapt those for the synchronous interleaved scheme in order to take advantage of the efficiency gain resulting from this scheme.

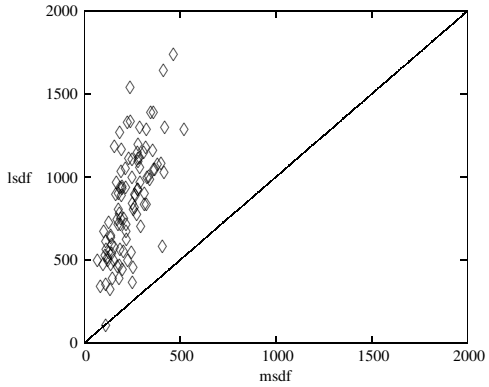


Figure 5.12: msdf vs lsdf in intervals

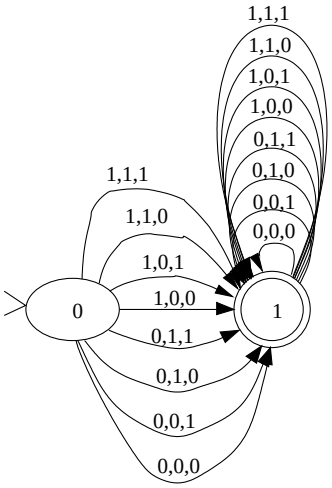


Figure 5.13: reduced minimal DFA for $\mathbb{Z}^3$, synchronous encoding scheme

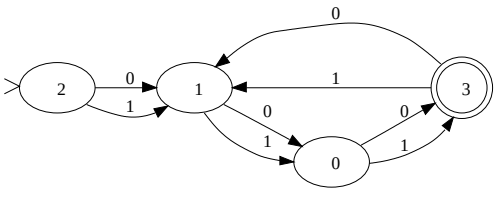


Figure 5.14: reduced minimal DFA for $\mathbb{Z}^3$, synchronous interleaved encoding scheme

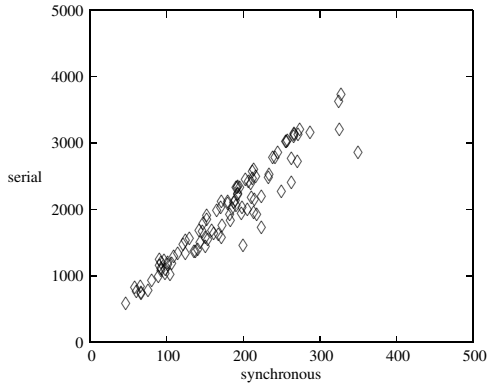


Figure 5.15: synchronous vs serial in equations

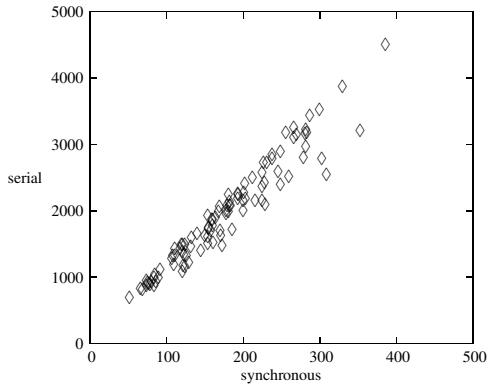


Figure 5.16: synchronous vs serial in inequations

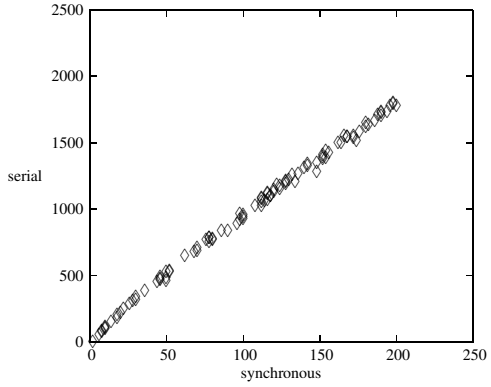


Figure 5.17: synchronous vs serial in congruences with modulo prime to basis

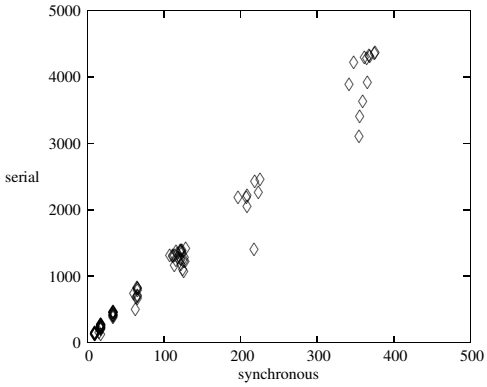


Figure 5.18: synchronous vs serial in congruences with modulo power of the basis

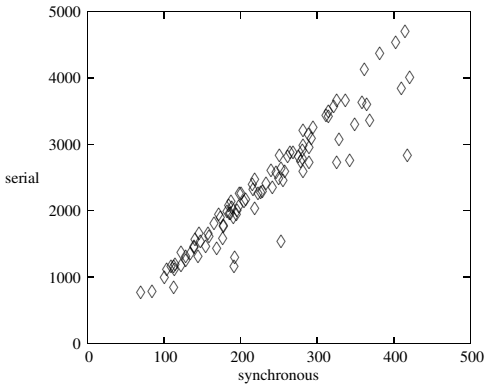


Figure 5.19: synchronous vs serial in intervals

Part II

From Automata to Formula

Chapter 6

Over-Approximation : Affine Hull of NDDs

In this chapter, we present a first approach for extracting information about sets represented by NDDs. We describe algorithms that, given a reduced NDD in strong normal form representing a set S of integer vectors, generate the affine hull of the set S over either $\mathbb{Q}$ or $\mathbb{Z}$. Most of those results appeared in [Lat05b]. The algorithms we present take as input a reduced (possibly nondeterministic) NDD in strong normal form using the synchronous encoding scheme.

6.1 Triangular Sets

The algorithms presented in this chapter manipulate intensively vector spaces over $\mathbb{Q}$, $\mathbb{Z}$-modules and $\mathbb{Z}_m$-modules. In order to have more efficient procedures, we maintain sets of generators in a particular form : the *triangular form* [MS05a]. For a nonzero vector $\mathbf{x}$, we call i the *leading index* of $\mathbf{x}$ and $\mathbf{x}[i]$ the *leading entry* of $\mathbf{x}$ if $\mathbf{x}[i] \neq 0$ and $\mathbf{x}[j] = 0$ for $j \in \{1, \ldots, i-1\}$. A set of nonzero vectors T is *triangular* if the leading entries of all vectors in T are positive and for all distinct vectors $\mathbf{x}, \mathbf{x}' \in T$, the leading indices of $\mathbf{x}$ and $\mathbf{x}'$ are distinct. Intuitively, a set is triangular if the vectors are the columns of a column echelon matrix $\mathbf{A}$ with no zero-column, i.e. each column of $\mathbf{A}$ has a nonzero element and if $\mathbf{A}[k_i, i]$ and $\mathbf{A}[k_j, j]$ are the first nonzero element of the ith and jth columns respectively with $j > i$, then $k_j > k_i$. Note that the vectors belonging to a triangular set of integer vectors are necessarily linearly independent.

Efficient procedures for generating an integer basis in triangular form of a

vector space over $\mathbb{Q}$ or of a $\mathbb{Z}$-module given a set of integer generators are available from the current literature. In particular, we have the following results.

Proposition 101. *There exists an algorithm* GETTRIANGQBASIS *which, given a finite set $G \subseteq \mathbb{Z}^n$ as input, generates a triangular set $T \subseteq \mathbb{Z}^n$ such that*

$$\mathrm{lin}_{\mathbb{Q}}(G) = \mathrm{lin}_{\mathbb{Q}}(T).$$

The sizes of the components of vectors in T are bounded by $n \cdot (k + \log n)$ where k is the bound on the component size of vectors in G, and the time complexity of GETTRIANGQBASIS *is $\mathcal{O}(|G| \cdot n^2)$.*

Proof. Let $\mathbf{A}$ be the matrix whose columns are the vectors in G, and let $\mathbf{B}$ be the matrix obtained by applying general Gaussian elimination to $\mathbf{A}^{\mathtt{t}}$ (see [Sch86]). Let $\mathbf{C}$ be the matrix obtained when multiplying each row of $\mathbf{B}$ by the least common multiplier of the denominators of the elements in the row. The set T is then the columns of $\mathbf{C}^{\mathtt{t}}$. By construction, the nonzero vectors in T are linear combinations of the vectors in G and vice versa. □

Proposition 102. *There exists an algorithm* UPDATETRIANGQ *which, given a triangular set $T \subseteq \mathbb{Q}^n$ and a vector $\mathbf{x} \in \mathbb{Q}^n$, generates a triangular set T' and a vector $\mathbf{x}' \in \mathbb{Q}^n$ such that the following assertions are valid.*

- $\mathrm{lin}_{\mathbb{Q}}(T') = \mathrm{lin}_{\mathbb{Q}}(T \cup \{\mathbf{x}\})$.
- *If* $\mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}(T)$, *then* $T' = T$
- *If* $\mathbf{x} \notin \mathrm{lin}_{\mathbb{Q}}(T)$, *then* $T' = T \cup \{\mathbf{x}'\}$.
- *The time complexity of* UPDATETRIANGQ *is* $\mathcal{O}(n^2)$.

Proof. The algorithm relies on the property that the vectors in a triangular set are linearly independent over $\mathbb{Q}$.

The function is recursive. If $\mathbf{x} = \mathbf{0}$, the function returns $(T, \mathbf{0})$. Assume that $\mathbf{x} \neq \mathbf{0}$ and let i be the leading index of $\mathbf{x}$.

- If for all $\mathbf{g} \in T$, the leading index of $\mathbf{g}$ is not i, then the function returns the pair $(G \cup \{\mathbf{x}\}, \mathbf{x})$.
- Otherwise, let $\mathbf{g}$ be the vector of T whose leading index is i. The function returns UPDATETRIANGQ$(T, \mathbf{x} - \frac{\mathbf{x}[i]}{\mathbf{g}[i]} \cdot \mathbf{g})$.

Since the leading index of the argument $\mathbf{x}$ in the recursive calls is strictly increasing, there are at most n recursive calls and the complexity of the algorithm is immediate.

The correctness of the algorithm can be proved by induction on the value of the leading index of the argument $\mathbf{x}$, starting with n and decreasing down to 1. □

Proposition 103. *There exists an algorithm* GETTRIANGZBASIS *which, given a finite set $G \subseteq \mathbb{Z}^n$ as input, generates a triangular set $T \subseteq \mathbb{Z}^n$ such that*

$$\mathrm{lin}_{\mathbb{Z}}(G) = \mathrm{lin}_{\mathbb{Z}}(T).$$

The sizes of the components of vectors in T are bounded by $k \cdot n \cdot \log n$, where k is the bound on the component size of vectors in G, and the time complexity of GETTRIANGZBASIS *is $\mathcal{O}(|G| \cdot k \cdot n^3 \cdot \log n)$.*

Proof. It suffices to generate the matrix $\mathbf{A}$ whose columns are the vectors in the set G, and then to compute the Hermite form $\mathbf{H}$ of $\mathbf{A}$, i.e. the matrix $\mathbf{H}$ such that $\mathbf{H}$ is in Hermite form and $\mathbf{H} = \mathbf{AU}$ for some square integer matrix $\mathbf{U}$ whose determinant is 1. The general complexity is obtained by using the computation of the Hermite form given in [Sto00]. □

Remark 104. *Computing a basis is more efficient over $\mathbb{Q}$ than over $\mathbb{Z}$. This is due to the fact that given a $\mathbb{Z}$-module S and a set $G \subset S$ of linearly independent vectors over $\mathbb{Z}$, there does not generally exist a set $G' \subseteq S$ such that $G \cup G'$ is a basis of S. For example, take $S = \{(x, y) \in \mathbb{Z}^2 \mid x = y\}$ and $G = \{(2, 2)\}$.* □

Proposition 105. *There exists an algorithm* INLINEARHULLZ? *which, given a triangular set $T \subseteq \mathbb{Z}^n$ and a vector $\mathbf{x} \in \mathbb{Z}^n$, returns* **true** *if $\mathbf{x} \in \mathrm{lin}_{\mathbb{Z}}(T)$ and* **false** *otherwise.*

The time complexity of INLINEARHULLZ? *is $\mathcal{O}(n^2)$.*

Proof. The algorithm relies on the property that the vectors in a triangular set are linearly independent.

The algorithm is recursive. If $\mathbf{x} = \mathbf{0}$, it returns **true** . Assume that $\mathbf{x} \neq \mathbf{0}$ and let i be the leading index of $\mathbf{x}$.

- If there is no vector in T whose leading index is i, then it returns **false** .
- If there is a vector $\mathbf{y} \in T$ whose leading index is i, then there are two possibilities.

– If $\mathbf{y}[i]$ does not divide $\mathbf{x}[i]$, then it return **false**.

– If $\mathbf{y}[i]$ divides $\mathbf{x}[i]$, then the algorithm returns INLINEARHULLZ?(T, $\mathbf{x}'$), with $\mathbf{x}' = \mathbf{x} - \frac{\mathbf{x}[i]}{\mathbf{y}[i]} \cdot \mathbf{y}$.

□

Proposition 106. *There exists an algorithm* UPDATETRIANGZM *which, given a strictly positive integer m, a triangular set $T \subseteq \mathbb{Z}_m^n$ and a vector $\mathbf{x} \in \mathbb{Z}_m^n$, generates a triangular set $T' \subseteq \mathbb{Z}_m^n$ such that*

$$\mathrm{lin}_{\mathbb{Z}_m}(T') = \mathrm{lin}_{\mathbb{Z}_m}(T \cup \{\mathbf{x}\}).$$

The time complexity of UPDATETRIANGZM *is $\mathcal{O}(n^2 \cdot \log m)$.*

Proof. See Section 6.6.1 □

The following property will be required when proving the complexity of an algorithm which calls recursively the procedure UPDATETRIANGZM.

Proposition 107. *Assume that $m = p_1^{q_1} \dots p_t^{q_t}$ for distinct prime numbers $p_1, \dots, p_t$. Then a sequence of triangular sets $T_1, T_2, \dots, \subseteq \mathbb{Z}_m^n$ such that $T_k \neq T_{k+1}$ and $T_{k+1} = \text{UPDATETRIANGZM}(m, T_k, \mathbf{x}_k)$ has length at most $n \cdot (q_1 + \dots + q_t)$.*

Proof. See Section 6.6.1. □

Proposition 108. *Given a triangular set $T \subseteq \mathbb{Z}^n$ with $|T| = q$, and a vector $\mathbf{x}_0 \in \mathbb{Z}^n$, there exists an algorithm that generates a set of congruences $\mathbf{a}_i.\mathbf{x} \equiv_{m_i} b_i$, $i = 1, \dots, q$, and a set of equations $\mathbf{a}_i.\mathbf{x} = b_i$, $i = q+1, \dots, n$, where numbers are bounded by $\mathcal{O}(n \log n + nk)$, k being a bound on the size of the numbers in the vectors in T and $\mathbf{x}_0$, such that*

$$\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Z}}(T) \Leftrightarrow \bigwedge_{i=1,\dots,q} \mathbf{a}_i.\mathbf{x} \equiv_{m_i} b_i \wedge \bigwedge_{i=q+1,\dots,n} \mathbf{a}_i.\mathbf{x} = b_i,$$

and

$$\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Q}}(T) \Leftrightarrow \bigwedge_{i=q+1,\dots,n} \mathbf{a}_i.\mathbf{x} = b_i.$$

Proof. Without loss of generality, we may assume that the vectors of T are the columns of a matrix $\begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix}$ such that $\mathbf{B} \subseteq \mathbb{Z}^{q \times q}$ is not singular, i.e. there exists a matrix $\mathbf{B}^{-1}$ such that $\mathbf{B}^{-1}\mathbf{B} = \mathbf{I}_q$ where $\mathbf{I}_q$ is the identity matrix with q rows and columns.

So, we have for all $\mathbf{x} \in \mathbb{Z}^n$,

$$\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Z}}(T) \Leftrightarrow (\mathbf{x} - \mathbf{x}_0) = \begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \mathbf{c}, \text{ for some } \mathbf{c} \in \mathbb{Z}^q. \tag{6.1}$$

Since $\mathbf{B}$ is not singular, $\begin{bmatrix} \mathbf{B}^{-1} & \mathbf{0} \\ \mathbf{C}\mathbf{B}^{-1} & -\mathbf{I}_{n-q} \end{bmatrix}$ is not singular, and we have

$$\mathbf{x} - \mathbf{x_0} = \begin{bmatrix} \mathbf{B} \\ \mathbf{C} \end{bmatrix} \mathbf{c} \Leftrightarrow \begin{bmatrix} \mathbf{B}^{-1} & \mathbf{0} \\ \mathbf{C}\mathbf{B}^{-1} & -\mathbf{I}_{n-q} \end{bmatrix} (\mathbf{x} - \mathbf{x}_0) = \begin{bmatrix} \mathbf{I}_q \\ \mathbf{0} \end{bmatrix} \mathbf{c}. \tag{6.2}$$

Let $\frac{\mathbf{a}_i^{\mathrm{t}}}{m_i}$ denotes the ith row of $\begin{bmatrix} \mathbf{B}^{-1} & \mathbf{0} \\ \mathbf{C}\mathbf{B}^{-1} & -\mathbf{I}_{n-q} \end{bmatrix}$, with $\mathbf{a}_i \in \mathbb{Z}^n$ and $m_i \in \mathbb{N} \setminus \{0\}$. From (6.1) and (6.2), we deduce that $\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Z}}(T)$ iff there exist $k_1, \ldots, k_q \in \mathbb{Z}$ such that $\mathbf{a}_i.(\mathbf{x} - \mathbf{x}_0) = m_i \cdot k_i$ for all $i \in \{1, \ldots, q\}$ and $\mathbf{a}_i.(\mathbf{x} - \mathbf{x}_0) = 0$ for all $i \in \{q+1, \ldots, n\}$, i.e.

$$\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Z}}(T) \Leftrightarrow \bigwedge_{i=1,\ldots,q} \mathbf{a}_i.(\mathbf{x} - \mathbf{x}_0) \equiv_{m_i} 0 \wedge \bigwedge_{i=q+1,\ldots,n} \mathbf{a}_i.(\mathbf{x} - \mathbf{x}_0) = 0. \tag{6.3}$$

With a similar reasoning, we also deduce the following equivalence.

$$\mathbf{x} \in \mathbf{x}_0 + \mathrm{lin}_{\mathbb{Q}}(T) \Leftrightarrow \bigwedge_{i=q+1,\ldots,n} \mathbf{a}_i.(\mathbf{x} - \mathbf{x}_0) = 0. \tag{6.4}$$

Finally, it is well-known that the coefficients of $\mathbf{B}^{-1}$ are quotients of determinants of sub-matrices of $\mathbf{B}$. Since the determinant of $\mathbf{B}$ can be expressed as a sum of $n!$ products of n elements of $\mathbf{B}$, its size is bounded by $n \log n + kn$, and therefore, the sizes of the elements in $\mathbf{B}^{-1}$ are bounded by $\mathcal{O}(n \log n + nk)$, and the sizes of the elements in $\mathbf{C}\mathbf{B}^{-1}$ are bounded by $\mathcal{O}(n \log n + nk)$. □

6.2 Affine Hulls over $\mathbb{Q}$

In this section, we present an algorithm which takes as input a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ and generates the affine hull over $\mathbb{Q}$ of the set represented by $\mathcal{A}$. Note that we do not assume that $\mathcal{A}$ is deterministic. Also, since $\mathcal{A}$ is reduced and is an NDD, for all $q \in Q_{\mathtt{F}}$, $S_{\mathcal{A}}^q \neq \emptyset$.

We first present an algorithm based on [MS04] and similar to [Ler04a]. Then we present a more efficient algorithm which takes advantage of the special affine

transformation corresponding to transitions in NDDs. In addition, this more efficient version is also part of the more sophisticated algorithm for computing the affine hull over $\mathbb{Z}$.

Recall from Section 5.1 that $S^q_{\mathcal{A}}$ denotes the set of vectors whose encodings label paths from any $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ to q in the NDD $\mathcal{A}$, and $S_{\mathcal{A}}$ denotes the set of integer vectors represented by the NDD $\mathcal{A}$. In addition, we define the sets V_q, $q \in Q$, as the vector spaces over $\mathbb{Q}$ such that if $S^q_{\mathcal{A}} \neq \emptyset$, $\mathrm{aff}_{\mathbb{Q}}(S^q_{\mathcal{A}}) = \mathbf{x}+V_q$ for some $\mathbf{x} \in \mathbb{Q}^n$. Similarly, the set V is the vector space over $\mathbb{Q}$ such that $\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathbf{x} + V$ for some $\mathbf{x} \in \mathbb{Q}^n$. Finally, we define $l_{\min}$ as the smallest integer such that for all states q, if $S^q_{\mathcal{A}} \neq \emptyset$ then there exists an encoding w, with $0 < |w| \leq l_{\min}$, labeling a path from some initial state to q.

The vector space V is related to the vector spaces V_q, $q \in Q$, as follows.

Theorem 109. *Let $\mathbf{x}_q \in S^q_{\mathcal{A}}$ for all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$ and let $q' \in Q_{\mathtt{F}}$.*

$$\mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q\in Q_{\mathtt{F}}} V_q\right) = V.$$

Proof. By definition, we have

$$\mathbf{x}_{q'} + V = \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathrm{aff}_{\mathbb{Q}}(\bigcup_{q\in Q_{\mathtt{F}}} S^q_{\mathcal{A}}). \tag{6.5}$$

Thanks to Proposition 6 and by definition, we have

$$\begin{aligned}
\mathrm{aff}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} S^q_{\mathcal{A}}\right) &= \mathrm{aff}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} \mathrm{aff}_{\mathbb{Q}}(S^q_{\mathcal{A}})\right) \\
&= \mathrm{aff}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} \mathbf{x}_q + V_q\right) \\
&= \mathbf{x}_{q'} + \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) + V_q\right)
\end{aligned}$$

In addition, we have

$$\mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) + V_q\right) = \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q\in Q_{\mathtt{F}}} V_q\right).$$

Indeed, we prove the muutal inclusion.

- Suppose $\mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) + V_q\right)$. We have $\mathbf{x} = \sum_{q\in Q_{\mathtt{F}}} a_q \cdot (\mathbf{x}_q - \mathbf{x}_{q'} + \mathbf{y}_q)$ with $a_q \in \mathbb{Q}$ and $\mathbf{y}_q \in V_q$ for all $q \in Q_{\mathtt{F}}$. By definition of a linear hull, $\mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q\in Q_{\mathtt{F}}} V_q\right)$.

- Suppose $\mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q\in Q_{\mathtt{F}}} V_q\right)$. We have

$$\begin{aligned}\mathbf{x} &= \sum_{q\in Q_{\mathtt{F}}} a_q \cdot (\mathbf{x}_q - \mathbf{x}_{q'}) + \sum_{q\in Q_{\mathtt{F}}} b_q \mathbf{y}_q \\ &= \sum_{q\in Q_{\mathtt{F}}} a_q \cdot (\mathbf{x}_q - \mathbf{x}_{q'} + \frac{b_q}{a_q}\mathbf{y}_q)\end{aligned}$$

  with $a_q, b_q \in \mathbb{Q}$ and $\mathbf{y}_q \in V_q$ for all $q \in Q_{\mathtt{F}}$. Since V_q are vector spaces, $\frac{b_q}{a_q}\mathbf{y}_q) \in V_q$, and by definition, we have $\mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) + V_q\right)$.

We conclude that $V = \mathrm{lin}_{\mathbb{Q}}\left(\bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q\in Q_{\mathtt{F}}} V_q\right)$. □

6.2.1 A First Algorithm

Our first algorithm is achieved in three steps.

1. One computes for each state q a vector $\mathbf{x}_q$ such that $\mathbf{x}_q \in S^q_{\mathcal{A}}$. This can be done via a breadth first search exploration of the states according to which one visits at step k all states q such that the smallest nonempty path from an initial state to q is of length k, and if w labels a path of length k, one sets $\mathbf{x}_q$ equal to $\langle w\rangle_{r,n}$.

2. For each state q, one computes a triangular set of vectors T_q such that $\mathbf{x}_q + \mathrm{lin}_{\mathbb{Q}}(T_q) = \mathrm{aff}_{\mathbb{Q}}(S^q_{\mathcal{A}})$.

3. One computes the triangular set T via a call GETTRIANGQBASIS($\bigcup_{q\in Q_{\mathtt{F}}} T_q \cup \bigcup_{q\in Q_{\mathtt{F}}}(\mathbf{x}_q - \mathbf{x}_{q'})$) for some $q' \in Q_{\mathtt{F}}$ with $S^{q'}_{\mathcal{A}} \neq \emptyset$, and the algorithm returns $(T, \mathbf{x}_{q'})$.

Theorem 110.

$$\mathbf{x}_{q'} + \mathrm{lin}_{\mathbb{Q}}(T) = \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}).$$

Proof. This is a direct consequence of Proposition 6 and Theorem 109. □

We now explain how to compute the sets T_q with $q \in Q$, assuming that for each $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, $\mathbf{x}_q \in S^q_{\mathcal{A}}$.

The computation relies on the fact that the vector spaces V_q, with $q \in Q$, are the smallest vector spaces K_q, with $q \in Q$ satisfying the following constraints.

$$\langle \alpha \rangle_{r,n} \in \mathbf{x}_{q'} + K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \text{ with } q \in Q_{\mathtt{I}} \quad (\mathrm{C.}\mathbb{Q}.1)$$

$$r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'} \in K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \text{ with } S^q_{\mathcal{A}} \neq \emptyset \quad (\mathrm{C.}\mathbb{Q}.2)$$

$$K_q \subseteq K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \quad (\mathrm{C.}\mathbb{Q}.3)$$

We first show that the vector spaces V_q, with $q \in Q$, satisfy the constraints.

Lemma 111. *The vector spaces V_q, $q \in Q$, satisfy* $(\mathrm{C.}\mathbb{Q}.1)$.

Proof. By definition, $\langle \alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}$ and $S^{q'}_{\mathcal{A}} \subseteq \mathbf{x}_{q'} + V_{q'}$. □

Lemma 112. *The vector spaces V_q, $q \in Q$, satisfy* $(\mathrm{C.}\mathbb{Q}.2)$.

Proof. By definition, there exists $u_q \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ with $\langle u_q \rangle_{r,n} = \mathbf{x}_q$ for some initial state $q_{\mathtt{I}} \in Q_{\mathtt{I}}$. Therefore, by hypothesis, $u_q\alpha \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q')$, and so, $\langle u_q\alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}$. By definition of the encoding scheme, $\langle u_q\alpha \rangle_{r,n} = r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n}$, and therefore, $r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}$. Since $S^{q'}_{\mathcal{A}} \subseteq \mathbf{x}_{q'} + V_{q'}$, we conclude that $r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'} \in V_{q'}$. □

Lemma 113. *The vector spaces V_q, $q \in Q$, satisfy* $(\mathrm{C.}\mathbb{Q}.3)$.

Proof. Thanks to Propositions 8 and 9,

$$V_q = \mathrm{lin}_{\mathbb{Q}}(-\mathbf{x}_q + S^q_{\mathcal{A}}) \quad (6.6)$$

$$V_{q'} = \mathrm{lin}_{\mathbb{Q}}(-\mathbf{x}_{q'} + S^{q'}_{\mathcal{A}}) \quad (6.7)$$

Let $\mathbf{g} \in V_q$. By definition, there exists $\mathbf{y}_1, \ldots, \mathbf{y}_t \in -\mathbf{x}_q + S^q_{\mathcal{A}}$ and $a_1, \ldots, a_t \in \mathbb{Q}$ such that

$$\mathbf{g} = \sum_{i=1}^{t} a_i \mathbf{y}_i \text{ with } a_1, \ldots, a_t \in \mathbb{Q}. \quad (6.8)$$

By definition, for each $i \in \{1, \ldots, t\}$, there exists $u_i \in L_{\mathcal{A}}(q_i \to q)$ for some initial state $q_i \in Q_{\mathtt{I}}$, such that $\mathbf{y}_i = -\mathbf{x}_q + \langle u_i \rangle_{r,n}$. Therefore, for each $i \in \{1, \ldots, t\}$, $u_i\alpha \in L_{\mathcal{A}}(q_i \to q')$ and $\langle u_i\alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}$. By definition of the encoding scheme, $\langle u_i\alpha \rangle_{r,n} = r \cdot \langle u_i \rangle_{r,n} + \langle o\alpha \rangle_{r,n}$, and therefore,

$$r \cdot (\mathbf{y}_i + \mathbf{x}_q) + \langle o\alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}. \quad (6.9)$$

In addition, thanks to Lemma 112,

$$r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'} \in V_{q'}. \tag{6.10}$$

Combining (6.7), (6.9) and (6.10), we deduce that for each $i \in \{1, \ldots, t\}$, we have

$$r \cdot \mathbf{y}_i \in V_{q'}. \tag{6.11}$$

Therefore, from (6.8), we deduce that $\mathbf{g} \in V_{q'}$, and we conclude that $V_q \subseteq V_{q'}$. □

Now, we show that any sequence of vector spaces K_q, with $q \in Q$ satisfying the constraints $(\mathrm{C}.\mathbb{Q}.1)$, $(\mathrm{C}.\mathbb{Q}.2)$ and $(\mathrm{C}.\mathbb{Q}.3)$ are such that $V_q \subseteq K_q$ for all $q \in Q$.

Lemma 114. *Let K_q, $q \in Q$, be a sequence of vector spaces over $\mathbb{Q}$ satisfying $(\mathrm{C}.\mathbb{Q}.1)$. $(\mathrm{C}.\mathbb{Q}.2)$ and $(\mathrm{C}.\mathbb{Q}.3)$. For all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, $K_q \supseteq V_q$.*

Proof. By definition, for all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, we have $\mathbf{x}_q + V_q = \mathrm{aff}_{\mathbb{Q}}(S^q_{\mathcal{A}})$, and therefore, $V_q = \mathrm{lin}_{\mathbb{Q}}(-\mathbf{x}_q + S^q_{\mathcal{A}})$. So, it suffices to prove that for all $q \in Q$ and $\mathbf{x} \in S^q_{\mathcal{A}}$, $\mathbf{x} - \mathbf{x}_q \in K_q$ in order to prove that $V_q \subseteq K_q$. This is proved by induction on the length of the encodings $w \in (\Sigma^n_r)^+$ of $\mathbf{x}$ such that $w \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ for some $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ and $q \in Q$. If $|w| = 1$, then this is a direct consequence of the fact that K_q, $q \in Q$, satisfy $(\mathrm{C}.\mathbb{Q}.1)$. Suppose the property holds for encodings of length $k \geq 1$, and let $w_{k+1} \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$, with $|w_{k+1}| = k + 1$, $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ and $q \in Q$. By hypothesis, $w_{k+1} = w_k\alpha$ for some encoding $w_k \in (\Sigma^n_r)^+$ and symbol $\alpha \in \Sigma^n_r$. Let q_k be such that $w_k \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_k)$ and $(q_k, \alpha, q) \in \Delta$. Since K_q, $q \in Q$, satisfy $(\mathrm{C}.\mathbb{Q}.2)$, we have

$$r \cdot \mathbf{x}_{q_k} + \langle o\alpha \rangle_{r,n} - \mathbf{x}_q \in K_q. \tag{6.12}$$

By inductive hypothesis, we have $\langle w_k \rangle_{r,n} - \mathbf{x}_{q_k} \in K_{q_k}$, and since K_q, $q \in Q$ satisfy $(\mathrm{C}.\mathbb{Q}.3)$, $K_{q_k} \subseteq K_q$. So we have

$$\langle w_k \rangle_{r,n} - \mathbf{x}_{q_k} \in K_q. \tag{6.13}$$

Finally, since $w_{k+1} = w_k\alpha$, by definition of the encoding scheme, we have

$$\langle w_{k+1} \rangle_{r,n} = r \cdot \langle w_k \rangle_{r,n} + \langle o\alpha \rangle_{r,n}. \tag{6.14}$$

Since K_q is a vector space, we can combine (6.12), (6.13) and (6.14), we find that $\langle w_{k+1} \rangle_{r,n} - \mathbf{x}_q \in K_q$. □

Theorem 115. *The vector spaces V_q, $q \in Q$, are the smallest vector spaces satisfying $(\mathrm{C}.\mathbb{Q}.1)$, $(\mathrm{C}.\mathbb{Q}.2)$ and $(\mathrm{C}.\mathbb{Q}.3)$.*

Proof. Direct consequence of Lemmas 111, 112, 113 and 114. □

Thanks to Theorem 115, the computation of the triangular sets T_q is a least fixpoint computation. Initially, for all states $q \in Q$, the set of vectors T_q are empty. First, for all transitions $(q, \alpha, q') \in \Delta$ such that $q \in Q_{\mathtt{I}}$, one sets $T_{q'}$ so that $\langle\alpha\rangle_{r,n} - \mathbf{x}_{q'} \in \mathrm{lin}_{\mathbb{Q}}(T_{q'})$. This is achieved via a call to UPDATETRIANGQ($T_{q'}$, $\langle\alpha\rangle_{r,n}$). Similarly, for all transitions $(q, \alpha, q') \in \Delta$, one sets $T_{q'}$ so that $r \cdot \mathbf{x}_q + \langle o\alpha\rangle_{r,n} - \mathbf{x}_{q'} \in \mathrm{lin}_{\mathbb{Q}}(T_{q'})$. Finally, while $\mathrm{lin}_{\mathbb{Q}}(T_q) \not\subseteq \mathrm{lin}_{\mathbb{Q}}(T_{q'})$ for some states $q, q' \in Q$ with $(q, \alpha, q') \in \Delta$, one updates $T_{q'}$ so that the inclusion is satisfied. This is done by sequentially updating $T_{q'}$ via the calls UPDATETRIANGQ($T_{q'}$, $\mathbf{g}_1$), ..., UPDATETRIANGQ($T_{q'}$, $\mathbf{g}_k$), where $\{\mathbf{g}_1, \ldots, \mathbf{g}_k\} = T_Q$. Thanks to Proposition 102, if $(T'_q, \mathbf{x}') = \text{UPDATETRIANGQ}(T_{q'}, \mathbf{x})$, then $T_q \subseteq T'_q$, i.e. whenever a vector is added to T_q, it is never removed. So, the test $\mathrm{lin}_{\mathbb{Q}}(T_q) \not\subseteq \mathrm{lin}_{\mathbb{Q}}(T_{q'})$ will only be verified when T_q is modified. So, it suffices to store in a set W the states q for which T_q has been modified and check only for those states whether one has to modify the set $T_{q'}$ of the successors q'.

The formal algorithm is given in Fig 6.1. Note that this algorithm always terminates since each T_q is modified at most n times.

Theorem 116. *Let $\mathbf{x} \in \mathbb{Z}^n$ and $T \subseteq \mathbb{Q}^n$ such that $(T, \mathbf{x}) =$ QAFFINEHULL_1$(\mathcal{A})$, with* QAFFINEHULL_1 *given in Fig 6.1.*

We have $\mathbf{x} + \mathrm{lin}_{\mathbb{Q}}(T) = \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}})$.

The time complexity of QAFFINEHULL_1 *is $\mathcal{O}(|\Delta| \cdot n^4)$.*

Proof. By construction, the vector spaces $\mathrm{lin}_Q(T_q)$, $q \in Q$, satisfy $(\mathrm{C}.\mathbb{Q}.1)$, $(\mathrm{C}.\mathbb{Q}.2)$ and $(\mathrm{C}.\mathbb{Q}.3)$, and therefore, thanks to Theorem 115, $\mathrm{lin}_Q(T_q) \supseteq V_q$. Also, by construction, the sets T_q, $q \in Q$, are modified via calls to UPDATETRIANGQ(T_q, $\mathbf{y}$) such that

- $\mathbf{y} = \langle\alpha\rangle_{r,n} - \mathbf{x}_q$, for some $\alpha \in \Sigma_r^n$ and $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ with $(q_{\mathtt{I}}, \alpha, q) \in \Delta$,
- $\mathbf{y} = r \cdot \mathbf{x}_{q'} + \langle o\alpha\rangle_{r,n} - \mathbf{x}_q$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$, or
- $\mathbf{y} \in \mathrm{lin}_{\mathbb{Q}}(T_{q'})$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$.

Therefore, thanks to Proposition 102 and Lemmas 111, 112 and 113, one proves by induction on the number of modifications brought to the sets T_q that $T_q \subseteq V_q$.

For each $q \in Q$, T_q is modified at most n times. So, by inspection, there are at most $n \cdot |\Delta|$ states added to W. Thanks to Proposition 102, the time complexity of UPDATETRIANGQ is $\mathcal{O}(n^2)$, and therefore, the overall complexity is $\mathcal{O}(|\Delta| \cdot n^4)$. □

function QAFFINEHULL_1(NDD $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_\mathtt{I}, Q_\mathtt{F})$) : (**set of** vectors in $\mathbb{Q}^n$, integer vector)

1: **var** W : **set of** state;
2: q, q' : state;
3: $T_1, \ldots, T_{|Q|}, T, G$: **set of** vectors in $\mathbb{Q}^n$;
4: α : symbol;
5: w : word;
6: $\mathbf{g}, \mathbf{g}', \mathbf{x}_1, \ldots, \mathbf{x}_{|Q|}$: vector in $\mathbb{Q}^n$;
7: **begin**
8: **for each** $q \in Q$ **do** $T_q := \emptyset$;
9: **for each** $q \in Q$ with $S_\mathcal{A}^q \neq \emptyset$, **let** $\mathbf{x}_q \in S_\mathcal{A}^q$;
10: **for each** $q \in Q_\mathtt{I}$, $q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta$ **do**
11: **begin**
12: $(T, \mathbf{g}) :=$ UPDATETRIANGQ($T_{q'}$, $\langle\alpha\rangle_{r,n} - \cdot\mathbf{x}_{q'}$);
13: **if** $T \neq T_{q'}$ **then** $W := W \cup \{q'\}$;
14: $T_{q'} := T$;
15: **end**
16: **for each** $q, q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta \wedge S_\mathcal{A}^q \neq \emptyset$ **do**
17: **begin**
18: $(T, \mathbf{g}) :=$ UPDATETRIANGQ($T_{q'}$, $r \cdot \mathbf{x}_q + \langle o\alpha\rangle_{r,n} - \cdot\mathbf{x}_{q'}$);
19: **if** $T \neq T_{q'}$ **then** $W := W \cup \{q'\}$;
20: $T_{q'} := T$;
21: **end**

(...)

Figure 6.1: Function QAFFINEHULL_1

```
    (...)
22:         while W ≠ ∅ do
23:             begin
24:                 let q ∈ W;
25:                 W := W \ {q};
26:                 for each q' ∈ Q such that (q, α, q') ∈ Δ with α ∈ Σ_r^n do
27:                     for each g ∈ T_q do
28:                         begin
29:                             (T, g') := UpdateTriangQ(T_q', r · g);
30:                             if T ≠ T_q' then W := W ∪ {q'};
31:                             T_q' := T;
32:                         end
33:             end
34:         G := ∅;
35:         let q' ∈ Q_F;
36:         for each q ∈ Q_F do G := G ∪ T_q ∪ {x_q − x_q'};
37:         return (GetTriangQBasis(G), x_q')
38:     end
```

Figure 6.2: Function QAffineHull_1 (continued)

As mentioned in [MH04], the efficiency of QAFFINEHULL_1 can be improved thanks to the following observation. When modifying a triangular set $T_{q'}$ via a call to UPDATETRIANGQ($T_{q'}$, g), all vectors in $T_{q'}$ before the call are left unchanged, the only difference that might occur is that one vector $\mathbf{g}'$ could be added to $T_{q'}$. So, when a triangular set T_q is modified, it is sufficient to update with the new element of T_q (and not with all other vectors in T_q) the sets $T_{q'}$ for all states q' such that $(q, \alpha, q') \in \Delta$ for some symbol $\alpha \in \Sigma_r^n$. Consequently, since each set is modified at most n times, there are at most $|\Delta| \cdot n$ calls to UPDATETRIANGQ, and the time complexity is reduced by a factor n, i.e. the time complexity is $\mathcal{O}(|\Delta| \cdot n^3)$. In the next section, we show another improvement which allows to get rid of all calls to UPDATETRIANGQ.

6.2.2 An Improved Algorithm

We can improve the algorithm QAFFINEHULL_1 presented in the previous section based on the following property which is an extension of Lemma 113 holding thanks to the fact that $\mathcal{A}$ is reduced.

Lemma 117. *For all $q \in Q$, $V_q \subseteq V$.*

Proof. Since $\mathcal{A}$ is reduced, there exist a sequence of symbols $\alpha_1, \ldots, \alpha_k \in \Sigma_r^n$ and a sequence of states $q_1, \ldots, q_k, q_{k+1} \in Q$ such that

- $q = q_1$,
- $q_{k+1} \in Q_{\mathtt{F}}$, and
- for all $i \in \{1, \ldots, k\}$, $(q_i, \alpha_i, q_{i+1}) \in \Delta$.

Thanks to Lemma 113, $V_{q_i} \subseteq V_{q_{i+1}}$ for all $i \in \{1, \ldots, k\}$, and therefore

$$V_q \subseteq V_{q_{k+1}}. \tag{6.15}$$

Also, by definition, $\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}^{q_{k+1}}) \subseteq \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}})$, and

$$V_{q_{k+1}} \subseteq V. \tag{6.16}$$

Combining (6.15) and (6.16), we have $V_q \subseteq V$. □

Thanks to the previous property, we deduce that it is not necessary to compute at each individual state q a triangular set T_q and a vector $\mathbf{x}_q$ such that $\mathbf{x}_q + \mathrm{lin}_{\mathbb{Q}}(T_q) = \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}^q)$. One only needs to consider one element $\mathbf{x}_q$ per state and

function QAFFINEHULL(NDD $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$) : (**set of** vectors in $\mathbb{Q}^n$, integer vector)

1: q, q' : state;
2: G : **set of** vectors in $\mathbb{Q}^n$;
3: α : symbol;
4: $\mathbf{g}, \mathbf{g}', \mathbf{x}_1, \ldots, \mathbf{x}_{|Q|}$: vector in $\mathbb{Q}^n$;
5: **begin**
6: $G := \emptyset$;
7: **for each** $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$, **let** $\mathbf{x}_q \in S_{\mathcal{A}}^q$;
8: **for each** $q \in Q_{\mathtt{I}}$, $q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta$ **do**
9: $G := G \cup \{\langle \alpha \rangle_{r,n} - \mathbf{x}_{q'}\}$;
10: **for each** $q, q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta \wedge S_{\mathcal{A}}^q \neq \emptyset$ **do**
11: $G := G \cup \{r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \cdot \mathbf{x}_{q'}\}$;
12: **let** $q' \in Q_{\mathtt{F}}$;
13: **for each** $q \in Q_{\mathtt{F}}$ **do** $G := G \cup \{\mathbf{x}_q - \mathbf{x}_{q'}\}$;
14: **return** $(G, \mathbf{x}_{q'})$
15: **end**

Figure 6.3: Function QAFFINEHULL

one set of generators for the whole NDD. Practically, this means that in Fig 6.1, we can substitute all triangular sets T_q by a single set T, and remove the main **while** -loop at line 22. Consequently, we don't have to check directly whether a vector $\mathbf{g}$ is in $\mathrm{lin}_{\mathbb{Q}}(T)$ or not, as it is done via the call UPDATETRIANGQ(T_q, $\mathbf{g}$). We choose to remove completely calls to UPDATETRIANGQ. This decreases the time complexity by a factor n^2 at the expense of a larger set of generators (at most $|\Delta|$ elements compared to n). This choice is justified by the fact that it is more efficient to perform once a call to GETTRIANGQBASIS with a set G that $|G|$ calls to UPDATETRIANGQ. Even the call GETTRIANGQBASIS is not part of the algorithm because it is not always required to have a triangular set. Also, this gives more flexibility, and this will be useful in the sequel.

The algorithm QAFFINEHULL displayed in Fig 6.3 incorporates the above considerations.

Theorem 118. *Let $\mathbf{x}_F \in \mathbb{Z}^n$ and $G \subseteq \mathbb{Z}^n$ such that $(G, \mathbf{x}_F) = \text{QAFFINEHULL}(\mathcal{A})$. We have*

$$\mathbf{x}_F + \text{lin}_{\mathbb{Q}}(G) = \text{aff}_{\mathbb{Q}}(S_{\mathcal{A}}).$$

The number of elements in G is bounded by $\mathcal{O}(|\Delta| + |Q|)$, the sizes of the components of the vectors in G are bounded by $\mathcal{O}(l_{\min})$ and the time complexity of QAFFINEHULL *is $\mathcal{O}(n \cdot (|\Delta| + |Q|))$,*

Proof. See Section 6.6.2. □

Since the set G of vectors generated by the algorithm QAFFINEHULL can get fairly large, one might be interested in computing a triangular set T corresponding to G, since a triangular set has at most n elements.

Theorem 119. *There exists an algorithm* QAFFINEHULLT *which, given a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$, generates a vector $\mathbf{x}_F$ and a triangular set of vectors T such that*

$$\text{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathbf{x}_F + \text{lin}_{\mathbb{Q}}(T).$$

The time complexity of QAFFINEHULLT *is $\mathcal{O}(|\Delta| \cdot n^2)$.*

Proof. Thanks to Theorem 118, by applying the algorithm QAFFINEHULL with $\mathcal{A}$ as input, we compute, in time proportional to $\mathcal{O}(|\Delta| \cdot n)$, a pair $(G, \mathbf{x}_F)$ such that $\text{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathbf{x}_F + \text{lin}_{\mathbb{Q}}(G)$. The number of elements in G is bounded by Δ and the sizes of the components of vectors in G are bounded by $\mathcal{O}(|Q|)$.

According to Proposition 101, we can compute a triangular set T of at most n generators from the set G. The size of the numbers in T is then bounded by $\mathcal{O}(n \cdot (|\Delta| + \log n))$ and the time complexity for the call GETTRIANGQBASIS(G) is $\mathcal{O}(|\Delta| \cdot n^2)$.

So, the overall time complexity of QAFFINEHULLT is $\mathcal{O}(|\Delta| \cdot n^2)$. □

Finally, thanks to Proposition 108, we can compute a system of linear equations corresponding to the affine hull, as shown by the next theorem.

Theorem 120. *There exists an algorithm* QAFFINEHULLEQUATIONS *which, given a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$, generates a system of at most n linear equations $\mathbf{A}\mathbf{x} = \mathbf{b}$ such that*

$$\text{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} = \mathbf{b}\}.$$

The time complexity of QAFFINEHULLEQUATIONS *is $\mathcal{O}(|\Delta| \cdot n^2)$.*

Proof. Direct consequence of Theorem 119 and of Proposition 108. □

6.3 Affine Hulls over $\mathbb{Z}$

In this section, we give an algorithm for computing the affine hull over $\mathbb{Z}$ of the set represented by a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$. Without loss of generality, we assume that for all accepting states $q \in Q_{\mathtt{F}}$, $S_{\mathcal{A}}^q \neq \emptyset$.

Note first that in general, if $(G, \mathbf{x}_F) = \text{QAFFINEHULL}(\mathcal{A})$, with QAFFINEHULL described in Fig.6.3, the set $\mathbf{x}_F + \text{lin}_{\mathbb{Z}}(G)$ is not equal to $\text{aff}_{\mathbb{Z}}(S_{\mathcal{A}})$. This stems from the fact that Lemma 117 does not hold in the integer case, since in the case of a $\mathbb{Z}$-module M, the fact that $r \cdot \mathbf{g} \in M_q$ does not imply that $\mathbf{g} \in M_q$.

Throughout this section, the sets M_q, $q \in Q$, are the $\mathbb{Z}$-modules such that if $S_{\mathcal{A}}^q \neq \emptyset$, $\text{aff}_{\mathbb{Z}}(S_{\mathcal{A}}^q) = \mathbf{x}_q + M_q$ for some $\mathbf{x}_q \in S_{\mathcal{A}}^q$. Similarly, the set M is the $\mathbb{Z}$-module such that $\text{aff}_{\mathbb{Z}}(S_{\mathcal{A}}) = \mathbf{x} + M$ for some $\mathbf{x} \in S_{\mathcal{A}}$. Finally, we define $d_{\min}$ and $l_{\min}$ as follows. The integer $d_{\min}$ is the smallest integer such that from each state $q \in Q$ reachable from some initial state, there is a path from q to an accepting state labeled by w with $|w| \leq d_{\min}$. The integer $l_{\min}$ is defined as in Section 6.2 as the smallest integer such that for all states q, if $S_{\mathcal{A}}^q \neq \emptyset$ then there exists an encoding w, with $0 < |w| \leq l_{\min}$, labeling a path from some initial state to q.

There is a relation between the $\mathbb{Z}$-module M and the $\mathbb{Z}$-modules M_q similar to the relation holding between the vector space V and the vector spaces V_q given in Section 6.2.

Theorem 121. *Let $\mathbf{x}_q \in S_{\mathcal{A}}^q$ for all $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$ and let $q' \in Q_{\mathtt{F}}$.*

$$\text{lin}_{\mathbb{Z}}\left(\bigcup_{q \in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q \in Q_{\mathtt{F}}} M_q\right) = M.$$

Proof. The proof is similar to the proof of Theorem 109. □

6.3.1 A first Algorithm

We first present an algorithm similar to QAFFINEHULL_1 presented in Section 6.2.1. Basically, the difference is that in QAFFINEHULL_1 one computes a set of generators for the vector spaces V_q, for all states $q \in Q$, whereas in the following algorithm, one computes a set of generators for the $\mathbb{Z}$-modules M_q, for all states q. Note that in both cases, the sets are generated via a fixpoint computation.

There are three steps in the algorithm.

1. One computes for each state q a vector $\mathbf{x}_q$ such that $\mathbf{x}_q \in S^q_{\mathcal{A}}$. This can be done through a simple breadth first search exploration starting at the initial states as explained in Section 6.2.1.

2. For all states q, one computes a set G_q such that $\mathrm{lin}_{\mathbb{Z}}(G_q) = M_q$.

3. The algorithm returns the pair $(\mathbf{x}_{q'}, G)$ where $q' \in Q_{\mathtt{F}}$ with $S^q_{\mathcal{A}} \neq \emptyset$ and G is constructed as follows.

$$G = \bigcup_{q \in Q_{\mathtt{F}}} G_q \cup \bigcup_{q \in Q_{\mathtt{F}}} \mathbf{x}_q - \mathbf{x}_{q'}.$$

Theorem 122.

$$\mathbf{x}_{q'} + \mathrm{lin}_{\mathbb{Z}}(G) = \mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}).$$

Proof. This is a direct consequence of Proposition 6 and Theorem 121. □

We now explain how to compute the sets G_q with $q \in Q$.

The computation relies on the fact that the $\mathbb{Z}$-modules M_q, with $q \in Q$, are the smallest $\mathbb{Z}$-modules K_q, with $q \in Q$ satisfying the following constraints.

$$\langle \alpha \rangle_{r,n} \in \mathbf{x}_{q'} + K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \text{ with } q \in Q_{\mathtt{I}} \quad (\mathrm{C}.\mathbb{Z}.1)$$

$$r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'} \in K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \text{ with } S^q_{\mathcal{A}} \neq \emptyset \quad (\mathrm{C}.\mathbb{Z}.2)$$

$$\{r \cdot \mathbf{y} \mid \mathbf{y} \in K_q\} \subseteq K_{q'} \text{ for each } (q, \alpha, q') \in \Delta \quad (\mathrm{C}.\mathbb{Z}.3)$$

We first show that the $\mathbb{Z}$-modules M_q, with $q \in Q$, satisfy the constraints $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$.

Lemma 123. *The $\mathbb{Z}$-modules M_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.1)$.

Proof. By definition, $\langle \alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}$ and $S^{q'}_{\mathcal{A}} \subseteq \mathbf{x}_{q'} + M_{q'}$. □

Lemma 124. *The $\mathbb{Z}$-modules M_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.2)$.

Proof. Similar to the proof of Lemma 112. □

Lemma 125. *The $\mathbb{Z}$-modules M_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.3)$.

Proof. The proof is similar to the proof of Lemma 113, except that linear hull and affine hull are over $\mathbb{Z}$. □

Now, we show that the $\mathbb{Z}$-modules M_q are the smallest $\mathbb{Z}$-modules satisfying the constraints $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$.

Lemma 126. *Let K_q, $q \in Q$, be a sequence of $\mathbb{Z}$-modules satisfying $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$. For all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, $K_q \supseteq M_q$.*

Proof. The proof is similar to the proof of Lemma 114. □

Theorem 127. *The $\mathbb{Z}$-modules M_q, $q \in Q$, are the smallest $\mathbb{Z}$-modules satisfying the constraints $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$.*

Proof. Direct consequence of Lemmas 123, 124, 125 and 126. □

Thanks to Theorem 127, the computation of the sets G_q is a least fixpoint computation. Initially, for all states $q \in Q$, the set of vectors G_q are empty. First, for all transitions $(q, \alpha, q') \in \Delta$ such that $q \in Q_{\mathtt{I}}$, one adds $\langle \alpha \rangle_{r,n} - \mathbf{x}_{q'}$ to $G_{q'}$. Similarly, for all transitions $(q, \alpha, q') \in \Delta$, one adds $r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'}$ to $G_{q'}$.

Finally, while $\{r \cdot \mathbf{g} \mid \mathbf{g} \in G_q\} \not\subseteq \mathrm{lin}_{\mathbb{Z}}(G_{q'})$ for some states $q, q' \in Q$ with $(q, \alpha, q') \in \Delta$, one adds $\{r \cdot \mathbf{g} \mid \mathbf{g} \in G_q\}$ to $G_{q'}$. Note that in order to test for inclusion in the linear hull over $\mathbb{Z}$, one computes first a triangular set generating the same $\mathbb{Z}$-module, and this is done via the function GETTRIANGZBASIS.

The formal algorithm is given in Fig 6.4. The fact that this algorithm terminates is not as trivial as in the case of QAFFINEHULL_1. However, this is the case thanks to the fact that any sequence of $\mathbb{Z}$-modules $M_1, \ldots, M_k, \ldots \subseteq \mathbb{Z}^n$, with $M_i \subset M_{i+1}$, is bounded thanks to Proposition 17.

Theorem 128. *Let $\mathbf{x} \in \mathbb{Z}^n$ and $G \subseteq \mathbb{Q}^n$ such that $(G, \mathbf{x}) =$ ZAFFINEHULL_1$(\mathcal{A})$, with* ZAFFINEHULL_1 *given in Fig 6.4. We have*

$$\mathbf{x} + \mathrm{lin}_{\mathbb{Z}}(T) = \mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}).$$

Proof. Let $\mathbf{x}_q$, $q \in Q$ be the vector appearing in ZAFFINEHULL_1. By inspection, if $S^q_{\mathcal{A}} \neq \emptyset$, $\mathbf{x}_q \in S^q_{\mathcal{A}}$ and there exists a state q' in $Q_{\mathtt{F}}$ such that $\mathbf{x} = \mathbf{x}_{q'}$.

Thanks to Theorem 121,

$$\mathbf{x}_{q'} + \mathrm{lin}_{\mathbb{Z}} \left(\bigcup_{q \in Q_{\mathtt{F}}} M_q \cup \bigcup_{q \in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) \right) = \mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}). \tag{6.17}$$

So, it suffices to prove that for each $q \in Q_{\mathtt{F}}$, $\mathrm{lin}_{\mathbb{Z}}(G_q) = M_q$.

By construction, the $\mathbb{Z}$-modules $\mathrm{lin}_{\mathbb{Z}}(G_q)$, $q \in Q$, satisfy $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$, and therefore, thanks to Theorem 127, $\mathrm{lin}_{\mathbb{Z}}(G_q) \supseteq M_q$. Also, by construction, the sets G_q, $q \in Q$, are modified either by a call GETTRIANGZBASIS(G_q) which does not alter the linear hull, or by adding a vector $\mathbf{y}$ such that

function ZAFFINEHULL_1(NDD $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_\mathtt{I}, Q_\mathtt{F})$) : (**set of** vectors in $\mathbb{Z}^n$, integer vector)

1: **var** W : **set of** (state, **set of** vectors in $\mathbb{Z}^n$);
2: q, q' : state;
3: $G_1, \ldots, G_{|Q|}, G$: **set of** integer vector;
4: α : symbol;
5: $\mathbf{g}, \mathbf{x}_1, \ldots, \mathbf{x}_{|Q|}$: integer vector;
6: **begin**
7: **for each** $q \in Q$ with $S_\mathcal{A}^q \neq \emptyset$, **let** $\mathbf{x}_q \in S_\mathcal{A}^q$;
8: **for each** $q \in Q_\mathtt{I}$, $q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta$ **do**
9: **begin**
10: $W := W \cup \{q'\}$;
11: $G_{q'} := G_{q'} \cup \{\langle \alpha \rangle_{r,n} - \mathbf{x}_{q'}\}$;
12: **end**
13: **for each** $q, q' \in Q$, $\alpha \in \Sigma_r^n$ **such that** $(q, \alpha, q') \in \Delta \wedge S_\mathcal{A}^q \neq \emptyset$ **do**
14: **begin**
15: $W := W \cup \{q'\}$;
16: $G_{q'} := G_{q'} \cup \{r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \cdot \mathbf{x}_{q'}\}$;
17: **end**

(...)

Figure 6.4: Function ZAFFINEHULL_1

```
    (...)
18:         while W ≠ ∅ do
19:             begin
20:                 let q ∈ W;
21:                 W := W \ {q};
22:                 for each q' ∈ Q such that (q, α, q') ∈ Δ for some α ∈ Σ_r^n do
23:                     for each g ∈ G_q do
24:                         begin
25:                             G_q' := GetTriangZBasis(G_q');
26:                             if InLinearHullZ?(G_q', r · g) = false then
27:                                 begin
28:                                     G_q' := G_q' ∪ {r · g};
29:                                     W := W ∪ {q'};
30:                                 end
31:                         end
32:             end
33:         G := ∅;
34:         let q' ∈ Q_F;
35:         for each q ∈ Q_F do
36:             begin
37:                 G := G ∪ {x_q − x_q'};
38:                 G := G ∪ G_q;
39:             end
40:         return (G, x_q')
41:     end
```

Figure 6.5: Function ZAFFINEHULL_1 (continued)

- $\mathbf{y} = \langle\alpha\rangle_{r,n} - \mathbf{x}_q$, for some $\alpha \in \Sigma_r^n$ and $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ with $(q_{\mathtt{I}}, \alpha, q) \in \Delta$,
- $\mathbf{y} = r \cdot \mathbf{x}_{q'} + \langle o\alpha\rangle_{r,n} - \mathbf{x}_q$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$, or
- $\mathbf{y} = r \cdot \mathbf{g}$ with $\mathbf{g} \in G_{q'}$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$.

Therefore, thanks to Lemmas 123, 124 and 125, one proves by induction on the number of modifications brought to the sets G_q that $G_q \subseteq M_q$ for all $q \in Q$. □

6.3.2 An Improved Algorithm

In this section, we present a polynomial time algorithm computing the affine hull over $\mathbb{Z}$ of the set represented by an NDD.

Although the algorithm ZAFFINEHULL_1 always terminated, there is no bound on the number of computation steps. In [MS05a], they solve the problem by working with modular arithmetic. More precisely, they do not compute the affine relations over $\mathbb{Z}$ holding at some control locations, but the affine relation over $\mathbb{Z}_m$ for some given m. The difference is that the length of any sequence of $\mathbb{Z}_m$-modules $M_1, \ldots, M_k \ldots \subseteq \mathbb{Z}_m^n$ with $M_i \subset M_{i+1}$ is bounded by $n \log m$ [MS05a]. In the sequel, we show how to incorporate some modular arithmetic in the algorithm ZAFFINEHULL_1 and obtain a polynomial time algorithm computing the affine hull over $\mathbb{Z}$ of the set represented by an NDD.

As we already mentioned, the set returned by the call QAFFINEHULL($\mathcal{A}$), with QAFFINEHULL given in Fig 6.3, is not a set of generators of M. However, thanks to Lemma 125 and given the specification QAFFINEHULL, we have the following lemma.

Lemma 129. *Let* QAFFINEHULL *be the algorithm given in Fig 6.3.*

Let $G_{pre}, T_{pre} \subseteq \mathbb{Z}^n$ and $\mathbf{x} \in \mathbb{Z}^n$ such that $(G_{pre}, \mathbf{x})$ = QAFFINEHULL*($\mathcal{A}$) and T_{pre} is a basis over $\mathbb{Z}$ of G_{pre}. We have*

- *for all $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$, $M_q \subseteq \mathrm{lin}_{\mathbb{Z}}(T_{pre})$, and*
- *for all $\mathbf{g} \in T_{pre}$, $r^{d_{\min}}\mathbf{g} \in M$.*

Proof. See Section 6.6.3. □

Given Lemma 129, we modify the algorithm ZAFFINEHULL_1 presented in Section 6.3.1 as follows. Recall that in ZAFFINEHULL_1, one computes for all $q \in \mathbb{Q}$ with $S_{\mathcal{A}}^q \neq \emptyset$ the $\mathbb{Z}$-modules M_q such that $\mathbf{x}_q + M_q = \mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}^q)$ with $\mathbf{x}_q \in S_{\mathcal{A}}^q$. In the algorithm presented below, one computes the $\mathbb{Z}$-modules M'_q, for

all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$ such that the $\mathbb{Z}$-modules M'_q are the smallest $\mathbb{Z}$-modules such that $M_q \cup \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\} \subseteq M'_q$. By definition, we have

$$M'_q = \mathrm{lin}_{\mathbb{Z}}\left(\{\mathbf{x} - \mathbf{x}_q \mid \mathbf{x} \in S^q_{\mathcal{A}}\} \cup \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\}\right).$$

Theorem 121 still holds when substituting M'_q for M_q.

Theorem 130. *Let $\mathbf{x}_q \in S^q_{\mathcal{A}}$ for all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$ and let $q' \in Q_{\mathrm{F}}$.*

$$\mathrm{lin}_{\mathbb{Z}}\left(\bigcup_{q \in Q_{\mathrm{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q \in Q_{\mathrm{F}}} M'_q\right) = M.$$

Proof. Thanks to Theorem 121, we have

$$\mathrm{lin}_{\mathbb{Z}}\left(\bigcup_{q \in Q_{\mathrm{F}}} M_q \cup \bigcup_{q \in Q_{\mathrm{F}}} (\mathbf{x}_q - \mathbf{x}_{q'})\right) = M. \tag{6.18}$$

In addition, thanks to Lemma 129,

$$\{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\}) \subseteq M. \tag{6.19}$$

Combining (6.18) and (6.19), we have

$$\mathrm{lin}_{\mathbb{Z}}\left(\{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\} \cup \bigcup_{q \in Q_{\mathrm{F}}} M_q \cup \bigcup_{q \in Q_{\mathrm{F}}} (\mathbf{x}_q - \mathbf{x}_{q'})\right) = M. \tag{6.20}$$

By definition of M'_q, $q \in Q$, we conclude that

$$\mathrm{lin}_{\mathbb{Z}}\left(\bigcup_{q \in Q_{\mathrm{F}}} (\mathbf{x}_q - \mathbf{x}_{q'}) \cup \bigcup_{q \in Q_{\mathrm{F}}} M'_q\right) = M.$$

□

There are four steps in the modified algorithm.

1. One computes the set $G_{pre} \subseteq \mathbb{Z}^n$ with $(G_{pre}, \mathbf{x})$ = QAFFINEHULL($\mathcal{A}$), and one computes a set $T_{pre} \subseteq \mathbb{Z}^n$ such that T_{pre} is basis over $\mathbb{Z}$ of $\mathrm{lin}_{\mathbb{Z}}(G_{pre})$.

2. One computes for each state q a vector $\mathbf{x}_q$ such that $\mathbf{x}_q \in S^q_{\mathcal{A}}$. This can be done through a simple breadth first search exploration starting at the initial state, as explained in Section 6.2.1.

3. One computes the sets $G_q \subseteq \mathbb{Z}^n$, for all $q \in Q$, such that the sets G_q generate the $\mathbb{Z}$-modules M'_q.

4. The algorithm returns a pair $(G, \mathbf{x}_{q'})$ where $q' \in Q_{\mathtt{F}}$ and G is constructed as follows.
$$G = \bigcup_{q \in Q_{\mathtt{F}}} G_q \cup \bigcup_{q \in Q_{\mathtt{F}}} \mathbf{x}_q - \mathbf{x}_{q'}.$$

Theorem 131.
$$\mathbf{x}_{q'} + \mathrm{lin}_{\mathbb{Q}}(G) = \mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}).$$

Proof. This is a direct consequence of Proposition 6 and Theorem 130. □

We now explain how to compute the sets G_q for all $q \in Q$.

The computation relies on the fact that the $\mathbb{Z}$-modules M'_q, with $q \in Q$, are the smallest $\mathbb{Z}$-modules K_q, with $q \in Q$, satisfying the constraints $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$ given in Section 6.3.1, as well the following additional constraints.

$$K_q \subseteq \mathrm{lin}_{\mathbb{Z}}(T_{pre}) \text{ for each } q \in Q \qquad (\mathrm{C}.\mathbb{Z}.4)$$

$$K_q \supseteq \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\} \text{ for each } q \in Q \qquad (\mathrm{C}.\mathbb{Z}.5)$$

We first show that the $\mathbb{Z}$-modules M'_q, with $q \in Q$, satisfy the constraints.

Lemma 132. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.1)$.

Proof. Direct consequence of Lemma 123 and of the fact that $M_q \subseteq M'_q$ for all $q \in Q$. □

Lemma 133. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.2)$.

Proof. Direct consequence of Lemma 124 and of the fact that $M_q \subseteq M'_q$ for all $q \in Q$. □

Lemma 134. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.3)$.

Proof. Let $\{\mathbf{y}_1, \ldots, \mathbf{y}_t\}$ be a basis over $\mathbb{Z}$ of the $\mathbb{Z}$-module M_q, and assume that $T_{pre} = \{\mathbf{z}_1, \ldots, \mathbf{z}_k\}$. By definition, $M'_q = \mathrm{lin}_{\mathbb{Z}}(\{\mathbf{y}_1, \ldots, \mathbf{y}_t, r^{d_{\min}} \cdot \mathbf{z}_1, \ldots, r^{d_{\min}} \cdot \mathbf{z}_k\})$.

Let $\mathbf{y} \in M'_q$. By definition, we have

$$\mathbf{y} = \sum_{i=1}^{t} a_i \mathbf{y}_i + \sum_{i=1}^{k} b_i \cdot r^{d_{\min}} \cdot \mathbf{z}_i. \qquad (6.21)$$

For all $i \in \{1, \ldots, t\}$, $\mathbf{y}_i \in M_q$, and therefore, thanks to Lemma 125, $r \cdot \mathbf{y}_i \in M_{q'} \subseteq M'_{q'}$. Also, for all $i \in \{1, \ldots, k\}$, $r^{d_{\min}+1}\mathbf{z}_i \in \mathrm{lin}_Z(\{r^{d_{\min}} \cdot \mathbf{z}_1, \ldots, r^{d_{\min}} \cdot \mathbf{z}_k\}) \subseteq M_{q'}$. Since $M_{q'}$ is a $\mathbb{Z}$-module, we conclude that $r \cdot \mathbf{y} \in M'_{q'}$. □

Lemma 135. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.4)$.

Proof. By definition, $M'_q = \mathrm{lin}_{\mathbb{Z}}(M_q \cup \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\})$. Thanks to Lemma 129, for all $\mathbf{y} \in M_q$, $\mathbf{y} \in \mathrm{lin}_{\mathbb{Z}}(T_{pre})$, and therefore, for all $\mathbf{y}' \in M'_q$, $\mathbf{y} \in \mathrm{lin}_{\mathbb{Z}}(T_{pre})$. □

Lemma 136. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, satisfy* $(\mathrm{C}.\mathbb{Z}.5)$.

Proof. This is a direct consequence of the definition of M'_q. □

Lemma 137. *Let K_q, $q \in Q$, be a sequence of $\mathbb{Z}$-modules satisfying* $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$, $(\mathrm{C}.\mathbb{Z}.3)$, $(\mathrm{C}.\mathbb{Z}.4)$ *and* $(\mathrm{C}.\mathbb{Z}.5)$. *For all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, $K_q \supseteq M'_q$.*

Proof. Since the sequence of $\mathbb{Z}$-modules K_q, $q \in Q$, satisfies $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$ thanks to Lemma 126, for each $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$, we have

$$K_q \supseteq M_q. \tag{6.22}$$

Also, since the sequence of $\mathbb{Z}$-modules K_q satisfies $(\mathrm{C}.\mathbb{Z}.5)$, we have

$$K_q \supseteq \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\}. \tag{6.23}$$

Finally, by definition, we have $\mathrm{lin}_{\mathbb{Z}}(K_q) = K_q$, and therefore, combining (6.22) and (6.23), we have

$$K_q \supseteq \mathrm{lin}_{\mathbb{Z}}(M_q \cup \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\}). \tag{6.24}$$

The claim is then immediate since $M'_q = \mathrm{lin}_{\mathbb{Z}}(M_q \cup \{r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}\})$. □

Theorem 138. *The $\mathbb{Z}$-modules M'_q, $q \in Q$, are the smallest $\mathbb{Z}$-modules satisfying the constraints* $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$, $(\mathrm{C}.\mathbb{Z}.3)$, $(\mathrm{C}.\mathbb{Z}.4)$ *and* $(\mathrm{C}.\mathbb{Z}.5)$.

Proof. Direct consequence of Lemmas 132, 133, 134, 135, 136 and 137. □

Basically, the difference between the algorithm presented in Section 6.3.1 and its modified version as presented in this section, is that in the former, one computes the smallest $\mathbb{Z}$-modules satisfying $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$ and $(\mathrm{C}.\mathbb{Z}.3)$, whereas in the latter, one computes the smallest $\mathbb{Z}$-modules satisfying $(\mathrm{C}.\mathbb{Z}.1)$, $(\mathrm{C}.\mathbb{Z}.2)$, $(\mathrm{C}.\mathbb{Z}.3)$, $(\mathrm{C}.\mathbb{Z}.4)$and $(\mathrm{C}.\mathbb{Z}.5)$. Adding the constraints $(\mathrm{C}.\mathbb{Z}.4)$and $(\mathrm{C}.\mathbb{Z}.5)$ provides a bound on the fixpoint computation, as shown in the following lemma.

Lemma 139. *The lengths of the sequences of $\mathbb{Z}$-modules $K_1, \ldots, K_k, \ldots \subseteq \mathbb{Z}^n$, with $K_i \subset K_{i+1}$, for all $1 \leq i < k$, and $\mathrm{lin}_{\mathbb{Z}}(r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}) \subseteq K_i \subseteq \mathrm{lin}_{\mathbb{Z}}(T_{pre})$, for all i, are bounded by $n \cdot d_{\min} \cdot \log r$.*

Proof. Let $\mathbf{g}_1, \ldots, \mathbf{g}_p$ be the vectors in T_{pre}, and let $K_1, \ldots, K_k, \ldots \subseteq \mathbb{Z}^n$, with $K_i \subset K_{i+1}$, for all $1 \leq i < k$, and $\mathrm{lin}_{\mathbb{Z}}(r^{d_{\min}} \cdot \mathbf{g} \mid \mathbf{g} \in T_{pre}) \subseteq K_i \subseteq \mathrm{lin}_{\mathbb{Z}}(T_{pre})$, for all $1 \leq i \leq k$.

Since for all i, $K_i \subseteq \mathrm{lin}_{\mathbb{Z}}(T_{pre})$, for all $\mathbf{y}$ in K_i, we have

$$\mathbf{y} = \sum_{i=1}^{p} a_i \cdot \mathbf{g}_i \text{ with } a_i \in \mathbb{Z} \text{ for all } i \in \{1, \ldots, p\}. \tag{6.25}$$

Since T_{pre} is a basis over $\mathbb{Z}$, the coefficients a_i are unique.

Also, since $K_i \supseteq \mathrm{lin}_{\mathbb{Z}}(\{r^{d_{\min}} \cdot \mathbf{g}_1, \ldots, r^{d_{\min}} \cdot \mathbf{g}_p\})$, for all $a_1, \ldots, a_p \in \mathbb{Z}^n$,

$$\sum_{j=1}^{p} a_j \cdot \mathbf{g}_i \in K_i \text{ iff } \sum_{j=1}^{p} (a_j \mod r^{d_{\min}}) \cdot \mathbf{g}_j \in K_i. \tag{6.26}$$

Based on the above considerations, we associate to each K_i a triangular set $C_i \subseteq \mathbb{Z}^p_{r^{d_{\min}}}$ such that

$$\sum_{j=1}^{p} a_i \mathbf{g}_j \in K_i \text{ iff } (a_1 \mod r^{d_{\min}}, \ldots, a_p \mod r^{d_{\min}})^{\mathtt{t}} \in \mathrm{lin}_{\mathbb{Z}_{r^{d_{\min}}}}(C_i). \tag{6.27}$$

The sets C_i are constructed as follows. First, we construct C_1. While there is a vector $\mathbf{y} \in K_1$ with $\mathbf{y} = \sum_{i=1}^{p} a_i \mathbf{g}_i$, such that the vector $\mathbf{a} \in \mathbb{Z}^p$, with $\mathbf{a}[i] = a_i$, is not in the linear hull over $\mathbb{Z}_{r^{d_{\min}}}$ of C_1, then one sets C_1 equals to UPDATETRIANGZM($r^{d_{\min}}$, C_1, $\mathbf{a} \mod r^{d_{\min}}$). Thanks to Proposition 107, C_1 is modified at most $n \cdot d_{\min} \cdot \log r$ times. Given C_j, one initializes C_{j+1} to be equal to C_j, and then C_{j+1} is modified in the same way as C_1 is constructed via the successive calls to the function UPDATETRIANGZM. So, C_k is obtained by successive modifications of C_1 via UPDATETRIANGZM, and thanks to Proposition 107, there are at most $n \cdot d_{\min} \cdot \log r$ $\mathbb{Z}$-modules K_i in the sequence. □

In the actual algorithm ZAFFINEHULL, formalized in Fig. 6.6, we do not store the sets G_q but the corresponding triangular set C_q as presented in the proof of Lemma 139. Note that the function DISTANCETOFINAL takes a reduced NDD in strong normal form $\mathcal{A}$ as input and returns the smallest number $d_{\min}$ such that for each state q, there is a path of length smaller or equal to $d_{\min}$ from q to an accepting state.

function ZAFFINEHULL(NDD $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_\mathtt{I}, Q_\mathtt{F})$) : (**set of** integer vector, integer vector)

```
 1:  var W : set of state;
 2:      q, q' : state;
 3:      G_pre, {g_1, ..., g_p}, C, C_1, ..., C_|Q| : set of integer vector;
 4:      α : symbol;
 5:      c, x_1, ..., x_|Q| : integer vector;
 6:      d_min : integer;
 7:  begin
 8:      for each q ∈ Q with S^q_A ≠ ∅, let x_q ∈ S^q_A;
 9:      for each q ∈ Q do C_q := ∅;
10:      (G_pre, x_F) := QAFFINEHULL(A);
11:      {g_1, ..., g_p} := GETTRIANGZBASIS(G_pre);
12:      d_min := DISTANCETOFINAL(A);
13:      for each q ∈ Q_I, q' ∈ Q, α ∈ Σ^n_r such that (q, α, q') ∈ Δ do
14:          begin
15:              W := W ∪ {q'};
16:              Let c ∈ Z^p such that ⟨α⟩_{r,n} − x_q = Σ_{i=1}^p c[i] g_i;
17:              C_{q'} := UPDATETRIANGZM(r^{d_min}, C_{q'}, c mod r^{d_min});
18:          end
19:      for each q, q' ∈ Q, α ∈ Σ^n_r such that S^q_¬ ≠ ∅ ∧ (q, α, q') ∈ Δ do
20:          begin
21:              W := W ∪ {q'};
22:              Let c ∈ Z^p such that r · x_q + ⟨oα⟩_{r,n} − x_{q'} = Σ_{i=1}^p c[i] g_i;
23:              C_{q'} := UPDATETRIANGZM(r^{d_min}, C_{q'}, c mod r^{d_min});
24:          end

    (...)
```

Figure 6.6: Function ZAFFINEHULL

```
    (...)
25:     while W ≠ ∅ do
26:         begin
27:             let q ∈ W;
28:             W := W \ {q};
29:             for each q' ∈ Q such that (q, α, q') ∈ Δ for some α ∈ Σ_r^n do
30:                 for each c ∈ C_q do
31:                     begin
32:                         C := UpdateTriangZm(r^{d_min}, C_{q'}, r · c);
33:                         if C ≠ C_{q'} do W := W ∪ {q'};
34:                         C_{q'} := C;
35:                     end
36:         end
37:     C := ∅;
38:     for each q ∈ Q_F, c ∈ C_q do C := UpdateTriangZm(r^{d_min}, C, c);
39:     G := ∅;
40:     for each c ∈ C do G := G ∪ {Σ_{i=1}^p c[i] g_i};
41:     for each i ∈ {1, ..., p} do G := G ∪ {r^{d_min} g_i};
42:     let q' ∈ Q_F;
43:     for each q ∈ Q_F do G := G ∪ {x_q − x_{q'}};
44:     return (G, x_{q'})
45: end
```

Figure 6.7: Function ZAFFINEHULL (continued)

Theorem 140. *Let $\mathbf{x}_F \in \mathbb{Z}^n$, $G \subseteq \mathbb{Z}^n$ such that $(G, \mathbf{x}_F) = \text{ZAFFINEHULL}(\mathcal{A})$, with* ZAFFINEHULL *displayed in Fig. 6.6. We have*

$$\mathbf{x}_F + \text{lin}_{\mathbb{Z}}(G) = \text{aff}_{\mathbb{Z}}(S_{\mathcal{A}}).$$

The number of vectors in G is bounded by $|Q| + 2n$, the sizes of components of vectors in G are bounded by $\mathcal{O}(n \cdot \log n \cdot l_{\min} + d_{\min})$, and the time complexity of ZAFFINEHULL *is $\mathcal{O}(|\Delta| \cdot n^3 \cdot d_{\min}^2 \cdot \log^2 r)$.*

Proof. In this proof, we consider the values of the variables used in Fig. 6.6 at the end of the computation, i.e. at line 44 in Fig. 6.6. For all $q \in Q$ with $S_{\mathcal{A}}^q$, let G_q be defined as follows.

$$G_q = \{\sum_{i=1}^{p} \mathbf{c}[i] \cdot \mathbf{g}_i \mid \mathbf{c} \in C_q\} \cup \{r^{d_{\min}}\mathbf{g}_1, \ldots, r^{d_{\min}}\mathbf{g}_p\}. \tag{6.28}$$

We show that for all $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$, $\text{lin}_{\mathbb{Z}}(G_q) = M'_q$. Once this is proved, by inspection, the correctness is immediate since, thanks to Theorem 130,

$$\mathbf{x}_{q'} + \text{lin}_{\mathbb{Z}}\left(\bigcup_{q \in Q_{\mathtt{F}}} M'_q \cup \bigcup_{q \in Q_{\mathtt{F}}} (\mathbf{x}_q - \mathbf{x}_{q'})\right) = \text{aff}_{\mathbb{Z}}(S_{\mathcal{A}}). \tag{6.29}$$

By construction and by inspection of Fig. 6.6, the sequence of $\mathbb{Z}$-modules $\text{lin}_{\mathbb{Z}}(G_q)$, $q \in Q$, satisfies (C.$\mathbb{Z}$.1), (C.$\mathbb{Z}$.2), (C.$\mathbb{Z}$.3)(C.$\mathbb{Z}$.4) and (C.$\mathbb{Z}$.5). So, thanks to Lemma 137, we have

$$\text{lin}_{\mathbb{Z}}(G_q) \supseteq M'_q, \tag{6.30}$$

for all $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$.

By construction, the sets C_q, $q \in Q$, are modified by calls UPDATETRIANGZM($r^{d_{\min}}$, C_q, $\mathbf{c} \mod r^{d_{\min}}$), such that

- $\sum_{i=1}^{p} \mathbf{c}[i] \cdot \mathbf{g}_i = \langle\alpha\rangle_{r,n} - \mathbf{x}_q$, for some $\alpha \in \Sigma_r^n$ and $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ with $(q_{\mathtt{I}}, \alpha, q) \in \Delta$,
- $\sum_{i=1}^{p} \mathbf{c}[i] \cdot \mathbf{g}_i = r \cdot \mathbf{x}_{q'} + \langle o\alpha\rangle_{r,n} - \mathbf{x}_q$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$, or
- $\mathbf{c} = r \cdot \mathbf{c}'$ with $\mathbf{c}' \in C_{q'}$ for some $q' \in Q$, $\alpha \in \Sigma_r^n$ with $(q', \alpha, q) \in \Delta$.

Therefore, thanks to Proposition 106 and Lemmas 132, 133 and 134, one proves by induction on the number of modifications brought to the sets C_q, $q \in Q$, that the following inclusion relation is always satisfied for all $q \in Q$,

$$\{\sum_{i=1}^{p} \mathbf{c}[i] \cdot \mathbf{g}_i \mid \mathbf{c} \in C_q\} \subseteq M'_q. \tag{6.31}$$

So, by inspection and by definition of M'_q, we have

$$G_q \subseteq M'_q, \tag{6.32}$$

for all $q \in Q$ with $S^q_{\mathcal{A}} \neq \emptyset$.

Combining (6.30) and (6.32), we deduce that $\mathrm{lin}_{\mathbb{Z}}(G_q) = M'_q$.

By definition, there are at most n vectors in a triangular set over $\mathbb{Z}^n_m$, and therefore $|C| \leq n$. Similarly, a basis over $\mathbb{Z}$ of a $\mathbb{Z}$-module of $\mathbb{Z}^n$ has at most n elements, and therefore $p \leq n$. So, by inspection, the number of vectors in G is bounded by $|Q| + 2n$. Thanks to Theorem 118, the sizes of the components of the vectors in G_{pre} are bounded by $\mathcal{O}(l_{\min})$, and thanks to Proposition 103, the sizes of the components in the vectors $\mathbf{g}_1, \ldots, \mathbf{g}_p$ are bounded by $\mathcal{O}(l_{\min} \cdot n \cdot \log n)$. Also, the vectors $\mathbf{x}_q$ can be computed via a breadth first search, and in this case, the sizes of the components are bounded by $\mathcal{O}(l_{\min})$ as explained in the proof of Theorem 118. So, by inspection, the sizes of the components of vectors in G are bounded by $\mathcal{O}(n \cdot \log n \cdot l_{\min} + d_{\min})$. Finally, thanks to Proposition 107, the sets C_q are modified at most $n \cdot d_{\min} \cdot \log r$, and therefore there are at most $|\Delta| \cdot n \cdot d_{\min} \cdot \log r$ calls to UPDATETRIANGZM. Thanks to Proposition 106, we conclude that the time complexity of ZAFFINEHULL is $\mathcal{O}(|\Delta| \cdot n^3 \cdot d^2_{\min} \cdot \log^2 r)$. □

Finally, exactly as we did for the linear hull over $\mathbb{Q}$, there exist algorithms generating a triangular set as well as a system of linear equations and congruences.

Theorem 141. *There exists an algorithm* ZAFFINEHULLT *which, given a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma^n_r, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$, generates a vector $\mathbf{x}_F$ and a triangular set of vectors T such that*

$$\mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}) = \mathbf{x}_F + \mathrm{lin}_{\mathbb{Z}}(T).$$

The time complexity of ZAFFINEHULLT *is $\mathcal{O}(|\Delta| \cdot |Q|^2 \cdot n^3 \cdot \log^2 r)$.*

Proof. Thanks to Theorem 140, by applying the algorithm ZAFFINEHULL with $\mathcal{A}$ as input, we generate in time proportional to $|\Delta| \cdot n^3 \cdot d^2_{\min} \cdot \log^2 r$ a pair $(G, \mathbf{x}_F)$

such that $\mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}) = \mathbf{x}_F + \mathrm{lin}_{\mathbb{Z}}(G)$ with $|G| \leq |Q| + 2n$. The sizes of the components of vectors in G are bounded by $\mathcal{O}(n \cdot \log n \cdot l_{\min} + d_{\min})$. Then thanks to Proposition 103, by applying the function GETTRIANGZBASIS to G, we can generate a basis T of $\mathrm{lin}_{\mathbb{Z}}(G)$ in time proportional to $\mathcal{O}(|G| \cdot k \cdot n^3 \cdot \log n)$, where $k = n \cdot \log n \cdot l_{\min} + d_{\min}$, and the size of the components of vectors in T are bounded by $k \cdot n \cdot \log n$.

Since $l_{\min}, d_{\min} \leq |Q|$, the overall complexity is therefore $\mathcal{O}(|\Delta| \cdot |Q|^2 \cdot n^3 \cdot \log^2 r)$. □

Finally, thanks to Proposition 108, we can compute a system of linear equations corresponding to the affine hull, as shown in the next theorem.

Theorem 142. *There exists an algorithm* ZAFFINEHULLEQUATIONS *which, given a reduced NDD in strong normal form $\mathcal{A} = (Q, \Sigma_r^n, \Delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$, generates a set of congruence relations $\mathbf{a}_i.\mathbf{x} \equiv_{m_i} b_i$, $i = 1, \ldots, k$, and a set of equations $\mathbf{a}_i.\mathbf{x} = b_i$, $i = k+1, \ldots, k+t$ such that*

$$\mathrm{aff}_{\mathbb{Z}}(S_{\mathcal{A}}) = \{\mathbf{x} \in \mathbb{Z}^n \mid \bigwedge_{i \in \{1,\ldots,k\}} \mathbf{a}_i.\mathbf{x} \equiv_{m_i} b_i \wedge \bigwedge_{i \in \{k+1,\ldots,k+t\}} \mathbf{a}_i.\mathbf{x} = b_i\}.$$

The time complexity of ZAFFINEHULLEQUATIONS *is $\mathcal{O}(|\Delta| \cdot |Q|^2 \cdot n^3 \cdot \log^2 r)$.*

Proof. This is a direct consequence of Theorem 141 and Proposition 108. □

6.4 Experimental Results

The algorithms QAFFINEHULL and ZAFFINEHULL presented in this chapter have been implemented within the LASH library [LAS]. Note that these algorithms have been slightly modified in order to use the synchronous interleaved encoding scheme, which significantly decreases the running time. We have taken $r = 2$ as encoding basis.

The time and memory used for the computation of the algorithms QAFFINEHULLT and ZAFFINEHULLT in a prototype implementation running on a Pentium-M at 1,5 GHz are given in the table below. The columns indicate successively the set on which the computation is performed, the number of components of the vectors in the set, the number of states in the corresponding NDD (with alphabet Σ_2), the values of $l_{\min}$ and $d_{\min}$ (see Theorems 118 and 140), and finally, the time and memory requirements for the computation of QAFFINEHULL and ZAFFINEHULL.

The sets used for testing our implementations of the algorithms are given in Fig.6.8. Note that all sets $S_1, \ldots, S_{12}$ are defined by a Boolean combination of several equations, inequations and congruence relations. In addition, $S_1, \ldots, S_6$ are $\mathbb{Z}$-affine modules, which is not the case of $S_7, \ldots, S_{12}$.

		$\mathcal{A}$			QAFFINEHULLT		ZAFFINEHULLT	
Set	n	Nb. States	$l_{\min}$	$d_{\min}$	Time (sec.)	Mem (Mb)	Time (sec.)	Mem (Mb)
S_1	7	64874	3	12	1.0	6.1	3.5	46.7
S_2	6	115727	2	15	1.6	10.4	4.6	64.5
S_3	6	287713	6	27	3.3	27.4	22.5	162.1
S_4	6	215685	4	4	3.3	22.5	10.8	123.4
S_5	10	281135	4	5	3.1	31.4	119.9	379.3
S_6	11	112754	2	5	2.3	13.1	10.9	183.4
S_7	7	279598	4	7	4.3	29.2	63.2	203.8
S_8	7	42067	5	10	0.8	4.3	6.4	30.6
S_9	6	54186	5	5	1.2	5.4	6.6	30.8
S_{10}	7	50580	5	6	0.7	5.1	7.2	36.7
S_{11}	6	52177	4	8	0.9	4.9	4.2	29.3
S_{12}	6	44920	6	7	1.0	4.4	4.5	25.4

In the above table, we note that in the sets considered, the values of $l_{\min}$ and $d_{\min}$ are small compared to $|Q|$. There exist sets for which the values of $l_{\min}$ and $d_{\min}$ have the same magnitude as $|Q|$. For example, the NDDs representing the sets $x \equiv_{2^k} 0$ in base 2 have k states and $l_{\min} \simeq d_{\min} \simeq k$. Our intuition is that whenever the characteristics numbers of a set (i.e. the constants appearing in a formula describing the set) are small then $l_{\min}$ and $d_{\min}$ are also small and our algorithms perform very well.

6.5 Conclusion

In this chapter, we have presented two algorithms, QAFFINEHULL and ZAFFINEHULL, that take a reduced (nondeterministic) NDD $\mathcal{A}$ as input and compute the affine hull over $\mathbb{Q}$ and over $\mathbb{Z}$ respectively of the set represented by $\mathcal{A}$ (note that the restriction that the NDDs be reduced is not constraining since any NDD can be reduced in linear time). More precisely, the algorithms generate a pair $(G, \mathbf{x}_F)$ with a finite set $G \subseteq \mathbb{Z}^n$ and a vector $\mathbf{x}_F \in \mathbb{Z}^n$ such that $\mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G)$

$$
\begin{aligned}
S_1 &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6) \in \mathbb{Z}^7 \mid 3x_0+2x_1-5x_2+6x_3-10x_4+ \\
&\quad 3x_5+2x_6=2 \wedge x_0+x_1+3x_2+2x_3+7x_4+15x_5-20x_6=2 \wedge \\
&\quad 10x_0+20x_1+30x_2=0\}
\end{aligned}
$$

$$
\begin{aligned}
S_2 &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 3x_0+2x_1-5x_2+6x_3-10x_4+3x_5=2 \\
&\quad \wedge x_0+x_1+3x_2+2x_3+7x_4+15x_5=2 \wedge 10x_0+20x_1+30x_2=0 \wedge \\
&\quad x_0+2x_1 \equiv_5 0\}
\end{aligned}
$$

$$
\begin{aligned}
S_3 &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 21x_0+3x_1-5x_2+2x_3+4x_4-x_5=24 \wedge \\
&\quad 5x_0+x_1-2x_2-2x_3+6x_4+3x_5=11 \wedge x_0 \equiv_{128} 0 \wedge x_0+x_1+x_2 \equiv_{49} 3\}
\end{aligned}
$$

$$
\begin{aligned}
S_4 &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 11x_0+5x_1+9x_2+19x_3+5x_4+6x_5 \equiv_{33} 0 \\
&\quad \wedge 2x_0+1x_1+3x_2+4x_3+6x_4+2x_5 \equiv_{33} 0 \wedge 5x_0+21x_1+1x_2+8x_3+ \\
&\quad 0x_4+x_5 \equiv_{33} 0\}
\end{aligned}
$$

$$
\begin{aligned}
S_5 &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8,x_9) \in \mathbb{Z}^{10} \mid x_0+x_1-x_2+3x_3+4x_4+ \\
&\quad x_5+5x_6+x_7+3x_8+x_9 \equiv_7 2 \wedge x_0-3x_2=3 \wedge 4x_3-5x_4=0 \wedge \\
&\quad x_7+x_8 \equiv_{20} 10 \wedge 10x_2-5x_9 \equiv_{16} 1 \wedge 2x_1+3x_5+x_6=12 \wedge x_7 \equiv_3 0\}
\end{aligned}
$$

$$
\begin{aligned}
S_6 &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8,x_9,x_{10}) \in \mathbb{Z}^{11} \mid 1x_0+2x_1-1x_4+1x_5=3 \\
&\quad \wedge 1x_2+2x_4-1x_9=2 \wedge 2x_1+1x_3+2x_5+1x_6-1x_7=5 \wedge 1x_1+1x_4+ \\
&\quad 2x_6+8x_8+1x_{10} \equiv_{13} 12 \wedge 2x_0+3x_2+1x_4+1x_7+8x_8+4x_9 \equiv_5 0\}
\end{aligned}
$$

$$
\begin{aligned}
S_7 &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6) \in \mathbb{Z}^7 \mid ((x_0+2x_1+3x_2-4x_5=2 \wedge 3x_0+3x_2+ \\
&\quad x_3+5x_4-6x_5+3x_6 \leq 10\,) \vee (4x_0+5x_2+2x_3-6x_4+3x_5+4x_6=2)) \\
&\quad \wedge x_0+x_1+x_3+3x_4+4x_5=4\}
\end{aligned}
$$

$$
\begin{aligned}
S_8 &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6) \in \mathbb{Z}^7 \mid (\,3x_0+x_1+8x_2+x_4+x_5+8x_6=2 \vee \\
&\quad 3x_0+x_1+8x_2+x_4+x_5+8x_6=27 \vee 3x_0+x_1+8x_2+x_4+x_5+8x_6=52 \\
&\quad \,) \wedge 4x_0+7x_1+2x_2-8x_3-x_4+4x_5=0 \wedge x_0 \geq 10 \wedge x_1 \geq 15\}
\end{aligned}
$$

$$
\begin{aligned}
S_9 &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 12x_0-9x_1+11x_3-2x_4 \equiv_{28} 5 \wedge 3x_1+x_2+ \\
&\quad 2x_3+5x_5 \equiv_{30} 1 \wedge x_0+2x_1+5x_2+4x_4+x_5=0 \wedge (x_0 \leq -10 \vee x_0 \geq 20) \\
&\quad \wedge (x_0+x_1+x_2+x_3+x_4+3x_5 \geq 0 \vee x_0+x_1+x_2+x_3+x_4+3x_5 \leq 50)\}
\end{aligned}
$$

$$
\begin{aligned}
S_{10} &= \{(x_0,x_1,x_2,x_3,x_4,x_5,x_6) \in \mathbb{Z}^7 \mid 3x_1+2x_2+x_3+2x_6 \equiv_{36} 2 \wedge x_1-6x_2+ \\
&\quad x_4+x_6=0 \wedge (x_1 \geq 10 \vee x_1 \leq 10) \wedge (x_2+x_3=20 \vee x_2+x_3 \leq -10) \wedge \\
&\quad (x_1+4x_2-10x_5 \leq 0)\}
\end{aligned}
$$

$$
\begin{aligned}
S_{11} &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 2x_0+3x_1+15x_2+11x_4+6x_5 \equiv_{20} 10 \wedge \\
&\quad x_0+x_1-6x_2+1x_3+x_4=0 \wedge (x_0+4x_1-x_2+x_3=10 \vee x_0+4x_1-x_2+x_3= \\
&\quad 18 \vee x_0+4x_1-x_2+x_3=32) \wedge (x_1+4x_2-10x_5 \leq 10)\}
\end{aligned}
$$

$$
\begin{aligned}
S_{12} &= \{(x_0,x_1,x_2,x_3,x_4,x_5) \in \mathbb{Z}^6 \mid 2x_0+3x_1+15x_2+11x_4+6x_5=3 \wedge \\
&\quad x_0+x_1-6x_2+x_3+x_4=0 \wedge (\,x_0+4x_1+6x_5 \equiv_4 3 \vee -x_2+x_3+3x_4 \equiv_{16} 10 \\
&\quad \vee x_0+5x_1+6x_2+3x_4=2 \vee x_0=0)\}
\end{aligned}
$$

Figure 6.8: Formulas of sets used in the experimental results

(resp. $\mathbf{x}_F + \mathrm{lin}_{\mathbb{Z}}(G)$) is the affine hull over $\mathbb{Q}$ (resp. $\mathbb{Z}$) of the set represented by $\mathcal{A}$. The size of the numbers manipulated in QAFFINEHULL (resp. ZAFFINEHULL) are bounded by $\mathcal{O}(|Q|)$ (resp. $\mathcal{O}(n \log n \cdot |Q|)$) and the time complexity is $\mathcal{O}(n \cdot (|\Delta| + |Q|))$ (resp. $\mathcal{O}(|\Delta| \cdot |Q|^2 \cdot n^3)$), where n is the number of components in the vectors, and Q and Δ are respectively the set of states and the transition relation of the input NDD $\mathcal{A}$.

Finally, note that the results we have presented also hold when considering reduced NDDs using the reverse synchronous encoding scheme. Indeed, those can be converted in linear time into a reduced (nondeterministic) NDDs using the synchronous encoding scheme by calling the function AUTO_REVERSE which simply flips the origins and the destinations of the transitions as well as the initial and the final states.

6.5.1 Related Work

An algorithm for computing the affine hull over $\mathbb{Q}$ of sets of positive vectors represented by NDDs can be obtained by adapting the algorithm in [MS04] which originally handles affine program. Affine programs are programs whose assignments are affine transformations, and in [MS04], the set of affine relations satisfied by the variables at some control locations is computed. Our algorithm QAFFINEHULL_1 is an adaptation of this algorithm by considering each transition in an NDD as an affine transformation. In addition, in [MS04], they consider only the incremental modifications brought to the triangular sets S_q while computing the least fixpoint, and, as mentioned in Section 6.2.1, this decreases the time complexity by a factor n. The time complexity of [MS04] adapted to NDD is $\mathcal{O}(|\Delta| \cdot n^3)$.

Another algorithm for computing the affine hull over $\mathbb{Q}$ of sets of positive vectors represented by NDDs is presented in [Ler04a]. This algorithm differs from QAFFINEHULL_1 as follows.

- Only encodings of positive integer vectors are handled in [Ler04a], i.e. NDDs represent subsets of $\mathbb{N}^n$ for some $n \in \mathbb{N}$.

- The encoding scheme is the reverse synchronous interleaved scheme. The fact that it deals with reverse encodings simply implies that one has to substitute the final states for the initial states and one has to reverse the transition, i.e. substitute (q', α, q) for (q, α, q'). The fact that the encoding is interleaved is more an implementation aspect and the algorithms QAFFINE-

HULL_1 and QAFFINEHULL can also be adapted to the synchronous interleaved scheme in a way similar to what is done in [Ler04b].

The complexity of the algorithm in [Ler04a] is identical to that of QAFFINEHULL_1, i.e. $\mathcal{O}(|\Delta| \cdot n^4)$.

Regarding the affine hull over $\mathbb{Z}$, there is nothing specific for sets represented by NDDs. In [Gra91], the set of affine relations as well as the congruence relations satisfied by the variable at some control locations in affine programs is computed. Although the computation always terminates, there is no bound on the number of execution steps required. The algorithm ZAFFINEHULL_1 presented in Section 6.2.1 can be roughly seen as an adaptation of [Gra91] to NDDs.

More recently, [MS05a] describes a polynomial time algorithm for computing affine relations over $\mathbb{Z}_m$, for some given m, satisfied by the variables at control locations in affine programs. While submitting this thesis, the authors have extended the algorithm presented in [MS05a] in order to deal with the more general case of finding affine relations as well as linear congruences in [MS05b]. The complexity of the proposed algorithm is polynomial and comparable to the complexity of our algorithm ZAFFINEHULL.

In the context of affine programs, we also want to mention [CH78]. The algorithm of [CH78] computes a system of linear inequations satisfied by the variables at some control location. This algorithm differs from the other algorithms mentioned aboved because the linear inequations computed only define over-approximation that does not have a single definition. This fact is directly related to the way the linear inequations are generated. Indeed, it uses a *widening operator* whose effect is to ensure that the algorithm terminates but whose precise effect is not explicit. Our intuition is that this algorithm could be also adapted to NDD with a more adequate widening operator and, in this context, it might lead to an algorithm computing the convex hull of the represented set.

Two other over-approximations for sets represented by NDDs have been proposed.

An algorithm computing the *semi-affine hull* (over $\mathbb{Q}$) of the set represented by an NDD has been introduced in [Ler03]. A *$\mathbb{Q}$-semi-affine space* is a finite union of $\mathbb{Q}$-affine spaces, and the semi-affine hull of a set S is the smallest $\mathbb{Q}$-semi-affine space including the set S. Given a set S, the semi-affine hull of S is included in the affine hull of S, and in general the inclusion is strict. For example, if one takes a finite set of vectors in $\mathbb{Q}^n$, the semi-affine hull of this set is the set itself, whereas the affine hull might be $\mathbb{Q}^n$ (if there are $n+1$ affinely independent vectors in the set). There are two drawbacks associated to the algorithm of [Ler03]

computing the semi-affine hulls of sets represented by NDDs. First, representing $\mathbb{Q}$-semi-affine spaces is expensive in general. For example, any finite set is a $\mathbb{Q}$-semi-affine space, and therefore, one might need a large number of vectors to describe a $\mathbb{Q}$-semi-affine-space. Secondly, the algorithm presented in [Ler03] is exponential in the number of states, and there is no indication that the algorithm might perform well when applied to practical examples.

Finally, an algorithm computing the convex hull over $\mathbb{Q}$ of the set represented by an NDD has been introduced in [FL05] and it is proved that the convex hull of any set representable by an NDD is a convex polyhedron. The method proposed in [FL05] is exponential and it is not clear how this algorithm would behave in practice.

6.6 Additional Proof Details

6.6.1 Proof of Propositions 106 and 107

In this section, we follow the main ideas presented in [MH04]. In order to handle the general case of a triangular set in $\mathbb{Z}_m$ where m is an arbitrary strictly positive integer, we first deal with the case when $m = p^q$ for some prime number p.

For a triangular set $T = \{\mathbf{y}_1, \ldots, \mathbf{y}_k\}$ in $\mathbb{Z}_{p^q}^n$, we define $\mathrm{rank}(T)$ as follows.

$$\mathrm{rank}(T) = \sum_i q_i + (n - k) \cdot q,$$

where q_i is such that $d_i p^{q_i}$ is the leading entry of $\mathbf{y}_i$ for some $d_i \in \mathbb{Z}$ with $\gcd(d_i, p)$. Clearly, we have

$$0 \leq \mathrm{rank}(T) \leq n \cdot q.$$

Proposition 143. *There exists an algorithm* UPDATETRIANGZPQ*, which, given a prime number p, a positive integer q, a triangular set $T \subseteq \mathbb{Z}_{p^q}^n$ and a vector $\mathbf{x} \in \mathbb{Z}_{p^q}^n$, such that if $T' =$* UPDATETRIANGZPQ$(p, q, T, \mathbf{x})$*, then the following assertions are valid.*

- *$T' \subseteq \mathbb{Z}_{p^q}^n$ and T' is triangular.*
- $\mathrm{lin}_{\mathbb{Z}_{p^q}}(T') = \mathrm{lin}_{\mathbb{Z}_{p^q}}(T \cup \{\mathbf{x}\})$.
- *If $T' \neq T$, then* $\mathrm{rank}(T') < \mathrm{rank}(T)$.

- *The time complexity of* UPDATETRIANGZPQ *is* $\mathcal{O}(n^2 \cdot q)$.

Proof. Recall that operations on elements of $\mathbb{Z}_{p^q}$ are carried out by using arithmetic modulo $\mathbb{Z}_{p^q}$.

Let i be the leading index of $\mathbf{x}$.

- If there is no vector in T whose leading index is also i, then it returns $T \cup \{\mathbf{x}\}$.
- If there is a vector $\mathbf{y}$ in T whose leading index is also i, and if $\mathbf{x}[i] = ap^b$ and $\mathbf{y}[i] = a'p^{b'}$, such that $\gcd(a, p) = 1$ and $\gcd(a', p) = 1$, there are two possibilities.
  - If $b' \leq b$, then there exists $c \in \{1, \ldots, p-1\}$ such that $\mathbf{x}[i] - cp^{b-b'} \cdot \mathbf{y}[i] \equiv_{p^q} 0$. In this case, the algorithm returns
    $$\text{UPDATETRIANGZPQ}(p, q, T, \mathbf{x} - cp^{b-b'}\mathbf{y}).$$
  - If $b' > b$, then the algorithm returns
    $$\text{UPDATETRIANGZPQ}(p, q, \mathbf{x} \cup T \setminus \{\mathbf{y}\}, \mathbf{y}).$$

See [MH04] for the correctness. □

We now turn to the general case, i.e. $m = \Pi_{i=1}^{t} m_i$, where $m_1, \ldots, m_t$ are pairwise relatively prime. The key observation is that every element of $x \in \mathbb{Z}_m$ is uniquely determined by the values $x_i = x \mod m_i$. That is, $x \equiv_m y$ if and only if $x \equiv_{m_i} y$ for $i = 1, \ldots, t$. This result is known as the Chinese Remainder Theorem. Interestingly, a corollary of this theorem is that given the remainders $x_1, \ldots, x_t$ with $0 \leq x_i < m_i$, we can generate a number x such that $x \equiv_{m_i} x_i$ for $i = 1, \ldots, t$. Indeed, thanks to Euclid's Algorithm, one can find in times $\mathcal{O}(\log m)$ numbers y_i, z_i such that $m_i \cdot y_i + \frac{m}{m_i} \cdot z_i = 1$. Let $s_i = \frac{m}{m_i} \cdot z_i \mod m$. By construction, for every $i, j \in \{1, \ldots, t\}$, we have

$$s_i \mod m_j = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ if } i \neq j \end{cases} \tag{6.33}$$

Finally, we compute x as $x = \left(\sum_{i=1}^{t} s_i \cdot x_i\right) \mod m$. From (6.33), one deduces that $x \equiv_{m_i} x_i$ for all $i \in \{1, \ldots, t\}$. The construction of x based on the remainders $x_1, \ldots, x_t$ is called the *Chinese remainder reconstruction* and is denoted $x =$

$x_1 * \ldots * x_t$. Note that the coefficients $s_1, \ldots, s_t$ are independent of $x_1, \ldots, x_t$ and need to be computed only once.

We extend the Chinese remainder reconstruction to vectors as follows. The vector $\mathbf{x} = \mathbf{x}_1 * \ldots * \mathbf{x}_k$ is such that $\mathbf{x}[i] = \mathbf{x}_1[i] * \ldots * \mathbf{x}_k[i]$. We can now state the main theorem.

Theorem 144. *Let $M \subseteq \mathbb{Z}_m^n$ be a $\mathbb{Z}_m$-module, and let $M_i = \{\mathbf{x} \mod m_i \mid \mathbf{x} \in M\}$ for $i \in \{1, \ldots, t\}$.*

1. *If G is a set of generators of M, then $G_i = \{\mathbf{x} \mod m_i \mid \mathbf{x} \in G\}$ is a set of generators of M_i.*
2. *For $i \in \{1, \ldots, t\}$, let $G_i = \{\mathbf{g}_1^{(i)}, \ldots, \mathbf{g}_n^{(i)}\}$ denotes a set of generators for M_i. Then $G = \{\mathbf{g_1}, \ldots, \mathbf{g}_n\}$ with $\mathbf{g}_j = \mathbf{g}_j^{(1)} * \ldots * \mathbf{g}_j^{(t)}$ is a set of generators of M.*

Proof. See [MH04]. □

Thanks to Theorem 144, we can generalize Proposition 143.

Proposition 145. *There exists an algorithm* UPDATETRIANGZM *which, given a strictly positive integer m, a triangular set $T \subseteq \mathbb{Z}_m^n$ and a vector $\mathbf{x} \in \mathbb{Z}_m^n$, generates a triangular set $T' \subseteq \mathbb{Z}_m^n$ such that*

$$\mathrm{lin}_{\mathbb{Z}_m}(T') = \mathrm{lin}_{\mathbb{Z}_m}(T \cup \{\mathbf{x}\}).$$

The time complexity of UPDATETRIANGZM *is $\mathcal{O}(n^2 \cdot \log m)$.*

Proof. Theorem 144 holds for any decomposition of m in pairwise relatively prime numbers and for any ordering of the elements in the sets G_i. In particular, we can choose prime powers $m_i = p_i^{q_i}$ and triangular sets G_i. So, Proposition 143 can be generalized to arbitrary m as follows. Compute $T^{(i)} = \{\mathbf{x} \mod m_i \mid \mathbf{x} \in T\}$ and $T^{(i)}_{\mathtt{new}} = \text{UPDATETRIANGZPQ}(p_i, q_i, T^{(i)}, \mathbf{x})$ for each $i \in \{1, \ldots t\}$.

- If $T^{(i)}_{\mathtt{new}} = T^{(i)}$ for all i, then return T.
- If $T^{(i)}_{\mathtt{new}} \neq T^{(i)}$ for some i, then return T' with $T' = \{\mathbf{g}_1, \ldots, \mathbf{g}_n\}$ such that $\mathbf{g}_j = \mathbf{g}_j^{(1)} * \ldots * \mathbf{g}_j^{(t)}$ where $\mathbf{g}_j^{(i)}$ is the vector of $T^{(i)}$ whose leading index is j or $\mathbf{0}$ if there is no vector whose leading index is j in $T^{(i)}$. □

As a corollary, we have the following proposition.

Proposition 146. *Assume that $m = p_1^{q_1} \dots p_t^{q_t}$ for pairwise different prime numbers p_i. Then every chain of triangular sets T_1, T_2, ... such that $T_k \neq T_{k+1}$ and $T_{k+1} = \text{UPDATETRIANGZM}(m, T_k, \mathbf{x}_k)$ has length at most $n \cdot (q_1 + \ldots + q_t)$.*

Proof. For all k and for all $i \in \{1, \ldots, t\}$, let $T_k^{(i)} = \{\mathbf{x} \mod m_i \mid \mathbf{x} \in T_k\}$. By construction, for all k, we have

$$\begin{aligned} &\text{rank}(T_{k+1}^{(i)}) \leq \text{rank}(T_k^{(i)}) \qquad \text{for all } i \in \{1, \ldots, t\}, \text{ and} \\ &\text{rank}(T_{k+1}^{(j)}) < \text{rank}(T_k^{(j)}) \qquad \text{for some } j \in \{1, \ldots, t\}. \end{aligned}$$

Also, by definition

$$0 \leq \text{rank}(T_k^{(i)}) \leq n \cdot q_i.$$

The claim is then immediate. □

6.6.2 Proof of Theorem 118

In this section, the set $G \subseteq \mathbb{Z}^n$ and the vector $\mathbf{x}_F \in \mathbb{Z}^n$ are such that $(G, \mathbf{x}_F) = \text{QAFFINEHULL}(\mathcal{A})$, where $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_\mathtt{I}, Q_\mathtt{F})$ is a reduced NDD in strong normal form and QAFFINEHULL is given in Fig. 6.3. Also, the vectors $\mathbf{x}_q \in \mathbb{Z}^n$ are those appearing in Fig. 6.3.

We first prove the following lemma.

Lemma 147. *For all $q \in Q$, $\mathbf{x} \in S_\mathcal{A}^q$, we have*

$$(\mathbf{x} - \mathbf{x}_q) \in \text{lin}_\mathbb{Q}(G).$$

Proof. We now show that for all states q, for all encodings w with $w \in L_\mathcal{A}(q_\mathtt{I} \to q)$ for some initial state $q_\mathtt{I} \in Q_\mathtt{I}$, we have $\langle w \rangle_{r,n} - \mathbf{x}_q \in \text{lin}_\mathbb{Q}(G)$. The proof is by induction on the length of w. By inspection, this holds if $|w| = 1$. Suppose this holds for all encodings of length smaller or equal to $k \geq 1$ and let $|w| = k + 1$ and q be such that $w \in L_\mathcal{A}(q_\mathtt{I} \to q)$ for some initial state $q_\mathtt{I} \in Q_\mathtt{I}$. By hypothesis, $w = w_k \alpha$ with $w_k \in (\Sigma_r^n)^+$ and $\alpha \in \Sigma_r^n$. Let q_k be such that $w_k \in L_\mathcal{A}(q_\mathtt{I} \to q_k)$ and $(q_k, \alpha, q) \in \Delta$. By inductive hypothesis, we have

$$\langle w_k \rangle_{r,n} - \mathbf{x}_{q_k} \in \text{lin}_\mathbb{Q}(G). \tag{6.34}$$

Also, by inspection, we have

$$r \cdot \mathbf{x}_{q_k} + \langle o\alpha \rangle_{r,n} - \mathbf{x}_q \in \text{lin}_\mathbb{Q}(G). \tag{6.35}$$

Combining (6.34) and (6.35), we have

$$r \cdot \langle w_k \rangle_{r,n} + \langle o\alpha \rangle_{r,n} - \mathbf{x}_q \in \mathrm{lin}_{\mathbb{Q}}(G). \tag{6.36}$$

By definition of the encoding scheme, $\langle w \rangle_{r,n} = r \cdot \langle w_k \rangle_{r,n} + \langle o\alpha \rangle_{r,n}$, and therefore, $\langle w \rangle_{r,n} - \mathbf{x}_q \in \mathrm{lin}_{\mathbb{Q}}(G)$. □

Theorem 148. *We have*

$$\mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G) = \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}).$$

The number of elements in G is bounded by $\mathcal{O}(|\Delta| + |Q|)$, the sizes of the components of the vectors in G are bounded by $\mathcal{O}(l_{\min})$ and the time complexity of QAFFINEHULL *is $\mathcal{O}(n \cdot (|\Delta| + |Q|))$,*

Proof. By inspection $\mathbf{x}_F \in S_{\mathcal{A}}$, and by definition

$$\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathbf{x}_F + V. \tag{6.37}$$

By inspection, for all $\mathbf{y} \in G$, there are three possibilities.

- $\mathbf{y} = \langle \alpha \rangle_{r,n} - \mathbf{x}_{q'}$ and there exists an initial state $q \in Q_{\mathtt{I}}$ such that $(q, \alpha, q') \in \Delta$. By definition, $\langle \alpha \rangle_{r,n} \in S_{\mathcal{A}}^{q'}$ and therefore, $\langle \alpha \rangle_{r,n} - \mathbf{x}_{q'} \in V_{q'}$. So, thanks to Lemma 117, $\mathbf{y} \in V$.

- $\mathbf{y} = r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'}$. By definition, there exists an encoding $w \in (\Sigma_r^n)^+$ with $\langle w \rangle_{r,n} = \mathbf{x}_q$ such that $w \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ for some initial state $q_{\mathtt{I}} \in Q_{\mathtt{I}}$. Therefore, $w\alpha \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q')$ and by definition of the encoding scheme, we have
$$r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} \in S_{\mathcal{A}}^{q'}. \tag{6.38}$$
So, we have $\mathbf{y} \in V_{q'}$ and thanks to Lemma 117, $\mathbf{y} \in V$.

- $\mathbf{y} = \mathbf{x}_q - \mathbf{x}_{q'}$, with $q, q' \in Q_{\mathtt{F}}$. So, by definition, $\mathbf{y} \in V$.

We deduce that $G \subseteq V$ and so, since V is a vector space over $\mathbb{Q}$, $\mathrm{lin}_{\mathbb{Q}}(G) \subseteq V$. Given (6.37), we conclude that

$$\mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G) \subseteq \mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}). \tag{6.39}$$

Let $\mathbf{x} \in S_{\mathcal{A}}^q$. By definition, there exist $w \in (\Sigma_r^n)^+$, $q_{\mathtt{I}} \in Q_{\mathtt{I}}$ and $q \in Q_{\mathtt{F}}$ such that $w \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$. Therefore, thanks to Lemma 147, we have

$$\mathbf{x} - \mathbf{x}_q \in \mathrm{lin}_{\mathbb{Q}}(G). \tag{6.40}$$

By inspection, $\mathbf{x}_F - \mathbf{x}_q \in G$, and therefore,

$$\mathbf{x} \in \mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G). \tag{6.41}$$

So, we deduce that $S_{\mathcal{A}} \subseteq \mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G)$. Thanks to Proposition 6, $\mathrm{aff}_{\mathbb{Q}}(\mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G)) = \mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G)$, and we conclude that

$$\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) \subseteq \mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G). \tag{6.42}$$

Combining (6.39) and (6.42), we find that $\mathrm{aff}_{\mathbb{Q}}(S_{\mathcal{A}}) = \mathbf{x}_F + \mathrm{lin}_{\mathbb{Q}}(G)$.

Clearly, the number of elements in G is bounded by $\mathcal{O}(|Q| + |\Delta|)$, and therefore, the time complexity is $\mathcal{O}(n \cdot (|Q| + |\Delta|))$. Finally, the vectors $\mathbf{x}_q$, $q \in Q$, can be computed via a breadth first search according to which one visits at step k all states q such that the smallest nonempty path from an initial state to q is of length k, and if w labels a path of length k, one sets $\mathbf{x}_q$ equal to $\langle w \rangle_{r,n}$. By definition of $l_{\min}$, there are $l_{\min}$ steps in the computation, and by definition of the encoding scheme, the sizes of the components of the vectors $\mathbf{x}_q$ are bounded by $\mathcal{O}(l_{\min})$. So, by inspection, the sizes of the components of the vectors in G are bounded by $\mathcal{O}(l_{\min})$. □

6.6.3 Proof of Lemma 129

In this section, the set $G_{pre} \subseteq \mathbb{Z}^n$ and the vector $\mathbf{x}_F \in \mathbb{Z}^n$ are such that $(G_{pre}, \mathbf{x}_F) = \textsc{QAffineHull}(\mathcal{A})$, where $\mathcal{A} = (Q, \Sigma_r^n, \Delta, Q_{\mathtt{I}}, Q_{\mathtt{F}})$ is a reduced NDD in strong normal form and QAffineHull is given in Fig. 6.3. Also, the vectors $\mathbf{x}_q \in \mathbb{Z}^n$ are those appearing in Fig. 6.3.

We first prove an auxiliary result.

Lemma 149. *For all $q \in Q$, $\mathbf{x} \in S_{\mathcal{A}}^q$, we have*

$$\mathbf{x} - \mathbf{x}_q \in \mathrm{lin}_{\mathbb{Z}}(G_{pre}).$$

Proof. The proof is the same as the proof of Lemma 147, it suffices to substitute $\mathrm{lin}_{\mathbb{Q}}(G)$ by $\mathrm{lin}_{\mathbb{Z}}(G_{pre})$. □

Lemma 150. *For all $q \in Q$ with $S_{\mathcal{A}}^q \neq \emptyset$, $M_q \subseteq \mathrm{lin}_{\mathbb{Z}}(G_{pre})$, and for all $\mathbf{g} \in G_{pre}$, $r^{d_{\min}}\mathbf{g} \in M$.*

Proof.

- Let $\mathbf{y} \in M_q$. By definition, $\mathbf{x}_q + M_q = \mathrm{aff}_{\mathbb{Z}}(S^q_{\mathcal{A}})$, and therefore,

$$\mathbf{y} = -\mathbf{x}_q + \sum_{i=1}^{k} a_i \mathbf{x}_i \,, \tag{6.43}$$

with $\sum_{i=1}^{k} a_i = 1$ and for all $i \in \{1, \ldots, k\}$, $a_i \in \mathbb{Z}$ and $\mathbf{x}_i \in S^q_{\mathcal{A}}$.

From (6.43), we have $\mathbf{y} = \sum_{i=1}^{k} a_i(\mathbf{x}_i - \mathbf{x}_q)$, and thanks to Lemma 149, for all $i \in \{1, \ldots, k\}$, we have $\mathbf{x}_i - \mathbf{x}_q \in \mathrm{lin}_{\mathbb{Z}}(G_{pre})$. So, by definition of the linear hull over $\mathbb{Z}$, we have $\mathbf{y} \in \mathrm{lin}_{\mathbb{Z}}(G_{pre})$.

- Let $\mathbf{y} \in G_{pre}$. There are three possibilities.
  - $\mathbf{y} = \langle \alpha \rangle_{r,n} - \mathbf{x}_{q'}$ and there exists an initial state $q \in Q_{\mathtt{I}}$ such that (q, α, q'). By definition, $\langle \alpha \rangle_{r,n}$ belongs to $S^{q'}_{\mathcal{A}}$ and therefore we have

$$\langle \alpha \rangle_{r,n} - \mathbf{x}_{q'} \in M_{q'}.$$

    By definition of $d_{\min}$, there exists a path from q' to a state $q_{\mathtt{F}} \in Q_{\mathtt{F}}$ labeled by w with $|w| \leq d_{\min}$. So, thanks to Lemma 125 and by definition of a $\mathbb{Z}$-module, we deduce that $r^{|w|} \cdot \mathbf{y} \in M_{q_{\mathtt{F}}}$, and thus

$$r^{d_{\min}} \cdot \mathbf{y} \in M_{q_{\mathtt{F}}}.$$

    Since $\mathbf{x}_{q_{\mathtt{F}}} + M_{q_{\mathtt{F}}} \subseteq \mathbf{x}_{q_{\mathtt{F}}} + M$, we conclude that $r^{d_{\min}} \cdot \mathbf{y} \in M$.
  - $\mathbf{y} = r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} - \mathbf{x}_{q'}$. By definition, there exists an encoding $u \in (\Sigma_r^n)^+$ with $\langle u \rangle_{r,n} = \mathbf{x}_q$ such that $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ for some initial state $q_{\mathtt{I}} \in Q_{\mathtt{I}}$. Therefore, $u\alpha$ labels a path from $q_{\mathtt{I}}$ to q' and thus $\langle u\alpha \rangle_{r,n}$ belongs to $S^{q'}_{\mathcal{A}}$. So, by definition of the encoding scheme, we have

$$r \cdot \mathbf{x}_q + \langle o\alpha \rangle_{r,n} \in S^{q'}_{\mathcal{A}}. \tag{6.44}$$

    So, we have $\mathbf{y} \in M_{q'}$. As shown previously, this implies that $r^{d_{\min}} \cdot \mathbf{y} \in M$.
  - $\mathbf{y} = \mathbf{x}_q - \mathbf{x}_{q'}$, with $q, q' \in Q_{\mathtt{F}}$. So, by definition, $\mathbf{y}$ belongs to M.

We conclude that for all $\mathbf{y} \in G$, $r^{d_{\min}} \cdot \mathbf{y}$ is in M. □

Chapter 7

Exact Formula : Integer Restrictions of Polyhedra

In this chapter, we present a method which, given the reduced minimal NDD using the synchronous encoding scheme $E_{S(r)}$ representing the integer solutions of a (convex) polyhedron P, generates a formula whose integer solutions are exactly the integer elements in P.

The choice of restricting the class of handled sets stems from the fact that there exist simple mathematical descriptions of the integer elements of polyhedra, and one of them is canonical. Also, thanks to [FL05], the convex hull over $\mathbb{Q}$ of any set represented by an NDD is a computable polyhedron. So, even if the initial NDD does not represent the integer elements of a polyhedron, it is still possible, in theory, to generate another NDD representing the convex hull over $\mathbb{Q}$ of the initial set, and apply on the resulting NDD the algorithms presented in this chapter.

By definition, a polyhedron corresponds to the solutions of a conjunction of inequations, and therefore, the integer elements of the polyhedron correspond to the integer solutions of the system of inequations. In addition, there is another representation for those sets, the extended Hilbert basis, i.e. a pair of finite sets of vectors, the *constants* and the *periods*, whose positive integer combinations generate the integer elements in the polyhedron. If the polyhedron is pointed, then, there exists a unique minimal extended Hilbert basis.

Interestingly, the sets of constants and periods of the minimal extended Hilbert basis corresponding to the integer elements of a pointed polyhedron P can be defined in Presburger arithmetic, extended with a predicate indicating the membership to $P \cap \mathbb{Z}^n$. So, a first approach to computing the extended Hilbert basis given an NDD representing $P \cap \mathbb{Z}^n$ is to construct the NDDs corresponding to the

sets of constants and periods by converting the defining formulas. This method is presented in the first section. It has experimentally been observed to be expensive.

We show in the second section interesting structural properties of reduced minimal NDDs representing the positive integer elements of polyhedra, and based on those properties, we develop an algorithm which, given the reduced minimal NDD representing the set of positive integer elements of a polyhedron, generates a system of linear inequations $\mathbf{Cx} \leq \mathbf{0}$ corresponding to the characteristic cone of this polyhedron.

Next, we show how to generalize those results to arbitrary signs. A polyhedron containing only positive integer elements is such that the encodings of all its elements have o as sign symbol. The generalization consists in considering polyhedra whose integer elements have all the same symbol $\alpha_{\texttt{sign}}$, which may be different from o.

In Section 7.4, we present an algorithm which, given the reduced minimal NDD representing the set of integer solutions of a pointed polyhedron P, generates the minimal extended Hilbert basis of the represented set. From this set, one then generates a formula defining $P \cap \mathbb{Z}^n$:

$$\bigvee_{i \in \{1,\ldots,t\}} \mathbf{C}(\mathbf{x} - \mathbf{x}_i) \leq \mathbf{0},$$

where $\{\mathbf{x}_1, \ldots, \mathbf{x}_t\}$ are the (finite) set of constants of the extended Hilbert basis.

The results presented in this chapter, except those in Section 7.1, only hold with the synchronous encoding scheme $E_{S(r)}$. This choice is motivated by the fact that NDDs are usually smaller when using the synchronous encoding scheme than when using the reverse synchronous encoding scheme, as shown in Section 5.5.1. Also, when dealing with an extension of NDDs, the Real Vector Automaton (RVA) [BRW98, BJW01], representing sets defined in the first order theory $\langle \mathbb{R}, +, <, \mathrm{Integer?} \rangle$, where Integer? is a one-place predicate testing for membership in $\mathbb{Z}$, the difference in size when using the synchronous compared to the reverse encoding scheme could even be more striking. Given the structural similarities between NDDs and RVAs, the results presented in this chapter could be applied to RVAs in the future.

7.1 Formula-based Generation of Basis

There is a simple solution for determining the constants and the periods of the extended Hilbert basis of the integer elements of a pointed polyhedron P repre-

sented by a NDD, independently of the choice of encoding scheme. Indeed, the sets of constants and periods can be defined via a formula in Presburger arithmetic extended with an additional predicate φ_P expressing the membership to $P \cap \mathbb{Z}^n$. Although we conclude this section on the fact that the formula-based approach is too expensive for the generation of the minimal extended Hilbert basis, the possibility to express the extended Hilbert basis in Presburger arithmetic gives some hints on the link between extended Hilbert bases and NDDs. The link is further detailed in the remaining sections.

We first present a formula φ_C such that for all $\mathbf{x} \in \mathbb{Z}^n$, $\varphi_C(\mathbf{x})$ holds if and only if $\mathbf{x} \in \text{char-cone}(P)$. Intuitively, a vector $\mathbf{x}$ is not in the characteristic cone if there exists a vector $\mathbf{y} \in P \cap \mathbb{Z}^n$ such that $\mathbf{x} + \mathbf{y} \notin P \cap \mathbb{Z}^n$.

$$\varphi_C(\mathbf{x}) =_{def} \neg(\exists \mathbf{y}, \mathbf{z} \in \mathbb{Z}^n)\,(\varphi_P(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x} + \mathbf{y} \wedge \neg\varphi_P(\mathbf{z})) \tag{7.1}$$

For convenience, we define φ_{C_0} as follows.

$$\varphi_{C_0}(\mathbf{x}) =_{def} \varphi_C(\mathbf{x}) \wedge \mathbf{x} \neq \mathbf{0}. \tag{7.2}$$

Similarly, we define the formula $\varphi_{\mathtt{cst}}$ and $\varphi_{\mathtt{per}}$ defining the membership to the sets of constants and of periods of the basis of $P \cap \mathbb{Z}^n$:

$$\varphi_{\mathtt{cst}}(\mathbf{z}) \;=_{def}\; \varphi_P(\mathbf{z}) \wedge \neg(\exists \mathbf{x}, \mathbf{y} \in \mathbb{Z}^n)(\varphi_P(\mathbf{x}) \wedge \varphi_{C_0}(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x} + \mathbf{y}) \tag{7.3}$$

$$\varphi_{\mathtt{per}}(\mathbf{z}) \;=_{def}\; \varphi_{C_0}(\mathbf{z}) \wedge \neg(\exists \mathbf{x}, \mathbf{y} \in \mathbb{Z}^n)(\varphi_{C_0}(\mathbf{x}) \wedge \varphi_{C_0}(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x} + \mathbf{y}) \tag{7.4}$$

Now, we show how to generate the NDDs $\mathcal{A}_C$, $\mathcal{A}_{\mathtt{cst}}$ and $\mathcal{A}_{\mathtt{per}}$ representing the sets of vectors satisfying respectively φ_C, $\varphi_{\mathtt{per}}$ and $\varphi_{\mathtt{cst}}$.

Based on Theorem 95, we can construct in time exponential in n a deterministic NDD $\mathcal{A}_{\mathbf{x}+\mathbf{y}=\mathbf{z}}$ in strong normal form, accepting the encodings of vectors $(\mathbf{x}, \mathbf{y}, \mathbf{z}) \in \mathbb{Z}^{3n}$ such that $\mathbf{z} = \mathbf{x} + \mathbf{y}$. The number of states of $\mathcal{A}_{\mathbf{x}+\mathbf{y}=\mathbf{z}}$ is exponential in n.

We can also construct the NDD $\mathcal{A}_{\neq 0}$ accepting the encodings of vectors $\mathbf{x} \in \mathbb{Z}^n$ such that $\mathbf{x} \neq \mathbf{0}$. The minimal reduced NDD accepting the set $\mathbb{Z}^2 \setminus \{\mathbf{0}\}$ in basis 2 is given in Fig.7.1.

The NDD $\mathcal{A}_C$ is obtained through the following automata-based operations involving $\mathcal{A}_P$ representing $P \cap \mathbb{Z}^n$, i.e. representing the predicate φ_P.

$$\begin{aligned}
\mathcal{A}_1 &= \text{NDD_COMPLEMENT}(\mathcal{A}_P)\\
\mathcal{A}_2 &= \text{AUTO_PRODUCT}(\mathcal{A}_{\mathbb{Z}^n}, \text{AUTO_PRODUCT}(\mathcal{A}_P), \mathcal{A}_1)\\
\mathcal{A}_3 &= \text{AUTO_INTERSECTION}(\mathcal{A}_{\mathbf{x}+\mathbf{y}=\mathbf{z}}, \mathcal{A}_2)\\
\mathcal{A}_4 &= \text{NDD_MULTI_PROJECTION}(\{n+1, \ldots 3n\}, \mathcal{A}_3)\\
\mathcal{A}_C &= \text{NDD_COMPLEMENT}(\mathcal{A}_4)
\end{aligned}$$

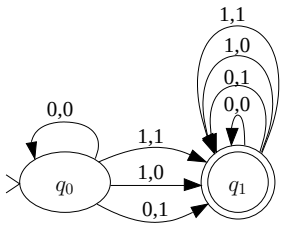


Figure 7.1: minimal NDD representing $S = \{(x, y) \in \mathbb{Z}^2 \mid (x, y) \neq (0, 0)\}$

The relationship between the automata $\mathcal{A}_1$, ..., $\mathcal{A}_3$, $\mathcal{A}_C$ and the formula (7.1) is given below.

$$\varphi_C(\mathbf{x}) =_{def} \neg\,(\exists \mathbf{y}, \mathbf{z} \in \mathbb{Z}^n)\ \underbrace{\underbrace{\underbrace{\left(\varphi_P(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x} + \mathbf{y} \wedge \underbrace{\neg\varphi_P(\mathbf{z})}_{\mathcal{A}_1}\right)}_{\mathcal{A}_3}}_{\mathcal{A}_4}}_{\mathcal{A}_C}$$

Thanks to Propositions 58, 59 and Theorems 77, 78 and 79, the size of the NDD $\mathcal{A}_C$ is $\mathcal{O}(2^{|Q|^2 \cdot 2^n})$.

The deterministic NDD $\mathcal{A}_{C_0}$ representing $\{\mathbf{x} \in \mathbb{Z}^n \mid \varphi_{C_0}(\mathbf{x})\}$ is generated from $\mathcal{A}_C$ as follows.

$$\mathcal{A}_{C_0} = \textsc{Auto\_Intersection}(\mathcal{A}_C, \mathcal{A}_{\neq \mathbf{0}})\,. \tag{7.5}$$

We construct a deterministic NDD representing the set of constants of the minimal extended Hilbert basis of $P \cap \mathbb{Z}^n$ as follows.

$$\begin{aligned}
\mathcal{A}_1 &= \textsc{Auto_Product}(\mathcal{A}_P, \textsc{Auto_Product}(\mathcal{A}_{C_0}, \mathcal{A}_{\mathbb{Z}^n})) \\
\mathcal{A}_2 &= \textsc{Auto_Intersection}(\mathcal{A}_{\mathbf{x}+\mathbf{y}=\mathbf{z}}, \mathcal{A}_1) \\
\mathcal{A}_3 &= \textsc{NDD_Multi_Projection}(\{1, \ldots 2n\}, \mathcal{A}_2) \\
\mathcal{A}_4 &= \textsc{NDD_Complement}(\mathcal{A}_3) \\
\mathcal{A}_{\mathtt{cst}} &= \textsc{Auto_Intersection}(\mathcal{A}_P, \mathcal{A}_4)
\end{aligned}$$

The relationship between the automata $\mathcal{A}_1, \ldots, \mathcal{A}_4, \mathcal{A}_{\mathtt{cst}}$ and the formula (7.3)

is given below.

$$\varphi_{\mathtt{cst}}(\mathbf{z}) =_{def} \varphi_P(\mathbf{z}) \wedge \neg\,(\exists \mathbf{x},\mathbf{y} \in \mathbb{Z}^n)\,\underbrace{\underbrace{\underbrace{\underbrace{(\varphi_P(\mathbf{x}) \wedge \varphi_{C_0}(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x}+\mathbf{y})}_{\mathcal{A}_2}}_{\mathcal{A}_3}}_{\mathcal{A}_4}}_{\mathcal{A}_{\mathtt{cst}}}.$$

Finally, we construct a deterministic NDD representing the set of periods of the minimal extended Hilbert basis as follows.

$$\begin{aligned}
\mathcal{A}_1 &= \text{AUTO_PRODUCT}(\mathcal{A}_{C_0}, \text{AUTO_PRODUCT}(\mathcal{A}_{C_0}, \mathcal{A}_{\mathbb{Z}^n}))\\
\mathcal{A}_2 &= \text{AUTO_INTERSECTION}(\mathcal{A}_{\mathbf{x}+\mathbf{y}=\mathbf{z}}, \mathcal{A}_1)\\
\mathcal{A}_3 &= \text{NDD_PROJECTION}(\{1,\ldots 2n\}, \mathcal{A}_2)\\
\mathcal{A}_4 &= \text{NDD_COMPLEMENT}(\mathcal{A}_3)\\
\mathcal{A}_{\mathtt{per}} &= \text{AUTO_INTERSECTION}(\mathcal{A}_{C_0}, \mathcal{A}_4)
\end{aligned}$$

The relationship between the automata $\mathcal{A}_1, \ldots, \mathcal{A}_4$, $\mathcal{A}_{\mathtt{per}}$ and the formula (7.3) is given below.

$$\varphi_{\mathtt{per}}(\mathbf{z}) =_{def} \varphi_{C_0}(\mathbf{z}) \wedge \neg\,(\exists \mathbf{x},\mathbf{y} \in \mathbb{Z}^n)\,\underbrace{\underbrace{\underbrace{\underbrace{(\varphi_{C_0}(\mathbf{x}) \wedge \varphi_{C_0}(\mathbf{y}) \wedge \mathbf{z} = \mathbf{x}+\mathbf{y})}_{\mathcal{A}_2}}_{\mathcal{A}_3}}_{\mathcal{A}_4}}_{\mathcal{A}_{\mathtt{per}}}.$$

Thanks to Propositions 58, 59 and Theorems 77, 78 and 79, the size of the NDD $\mathcal{A}_{\mathtt{cst}}$ and $\mathcal{A}_{\mathtt{per}}$ are $\mathcal{O}(2^{2^{|Q|^2 \cdot 2^n}})$.

We have tested several examples using the LASH [LAS]. It turns out that the method presented in this section is not efficient in practice, even when using the synchronous interleaved encoding scheme. For example, if φ_1 is the predicate testing membership into the set $S_1 = \{(x_1,x_2,x_3,x_4) \in \mathbb{Z}^4 \mid x_1 - x_2 - x_3 - 3x_4 \leq 2 \wedge -2x_1 + 3x_2 + 3x_3 - 5x_4 \leq 3\}$, computing $\mathcal{A}_C$ from the minimal NDD representing S_1 requires more than 512 megabytes of memory and takes more than two days, although the minimal NDD $\mathcal{A}_P$ (using the synchronous interleaved encoding scheme) has only 415 states. Even if $\mathcal{A}_C$ is provided, computing the periods or the constants of the minimal Hilbert basis also required more than 512 megabytes of memory and more than one day of computation. In the remaining sections,

we present a more efficient method for computing $\mathcal{A}_C$. In addition, it generates a system of linear inequations whose integer solutions are the integer elements satisfying φ_C. As a comparison, the method presented in the next section computes $\mathcal{A}_C$ and the minimal extended Hilbert basis from the minimal NDD representing S_1 in less than one second, and uses less than one megabyte of memory.

7.2 Synthesis of Formula for $\text{char-cone}(P)$ over the Natural Numbers

First, we highlight structural properties of NDDs representing positive integer elements of polyhedra. Then, we present an algorithm exploiting those properties which, given an NDD representing a set $P \cap \mathbb{N}^n$ where P is a polyhedron, synthesizes a formula corresponding to the characteristic cone of P.

Throughout this section, $\mathcal{A} = (\Sigma_r^n, Q, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ denotes the reduced minimal NDD using the synchronous encoding scheme $E_{S(r)}$ representing the set $S = P \cap \mathbb{N}^n$, where

$$P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\} \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\},$$

for some integer matrix $\mathbf{A} \in \mathbb{Z}^{m \times n}$ and integer vector $\mathbf{b} \in \mathbb{Z}^m$.

The inequations in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ are $\mathbf{a}_1.\mathbf{x} \leq b_1, \ldots, \mathbf{a}_m.\mathbf{x} \leq b_m$. Also, the characteristic cone of P is denoted by C, i.e.

$$C = \text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq 0\}.$$

Finally, $\mathbf{C}$ is an integer matrix such that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$ and such that no inequation in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ is redundant in $\mathbf{C}\mathbf{x} \leq \mathbf{0}$. The system of inequations $\mathbf{C}\mathbf{x} \leq \mathbf{0}$ can be generated by removing one by one the redundant inequations in $\mathbf{A}\mathbf{x} \leq \mathbf{0}$.

Note that many structural properties described in this section are valid only if $\mathcal{A}$ is reduced minimal. Also, since all elements of the represented set belong to $\mathbb{N}^n$, the sign symbols of all encodings of all elements in the represented set are o. Since $\mathcal{A}$ is reduced, this means that the first symbol of the words labeling paths rooted at $q_{\mathtt{I}}$ must be o.

Lemma 151. *For all non-empty words u labeling a path rooted at $q_{\mathtt{I}}$, we have*

- $\langle u \rangle_{r,n} \in \mathbb{N}^n$,

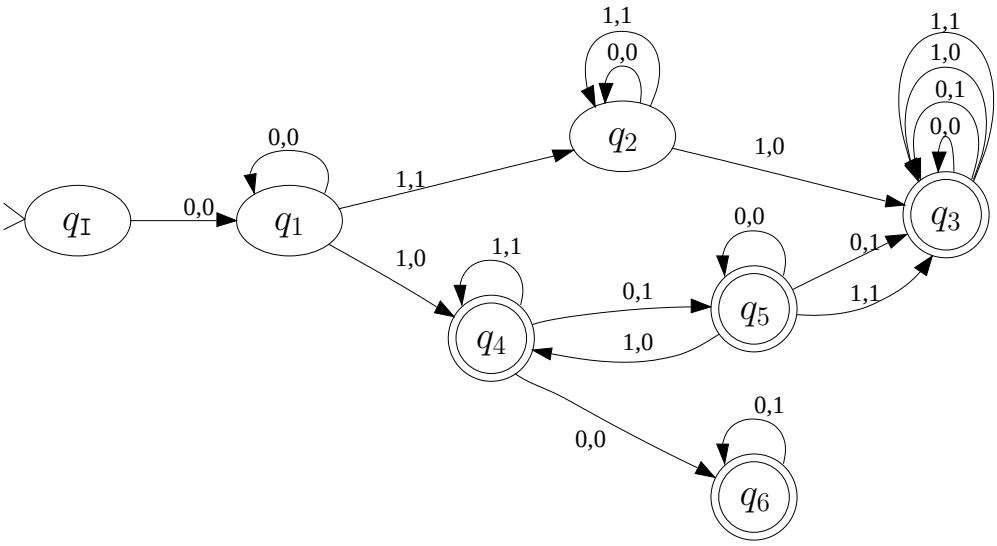


Figure 7.2: minimal reduced NDD $\mathcal{A}_{\mathtt{x}}$ representing $S_{\mathtt{x}}$

- $\langle ou \rangle_{r,n} = \langle u \rangle_{r,n}$, *and*
- $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, ou)$.

Proof. Let $q = \hat{\delta}(q_{\mathtt{I}}, u)$. Since $\mathcal{A}$ is reduced, there exists a word $w \in (\Sigma_r^n)^*$ such that $uw \in L(\mathcal{A})$, and by hypothesis, $\langle uw \rangle_{r,n} \in \mathbb{N}^n$. Therefore, by definition of the encoding scheme, the sign symbol of u is o, and therefore, $\langle u \rangle_{r,n} \in \mathbb{N}^n$ and $\langle ou \rangle_{r,n} = \langle u \rangle_{r,n}$.

Finally, since $\langle ou \rangle_{r,n} = \langle u \rangle_{r,n}$, for all words $w \in (\Sigma_r^n)^*$, $\langle ouw \rangle_{r,n} = \langle uw \rangle_{r,n}$, and so, $ouw \in L(\mathcal{A}) \Leftrightarrow uw \in L(\mathcal{A})$. So, since $\mathcal{A}$ is minimal, we conclude that $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, ou)$. □

Example 152. *We give in Fig.7.2 the reduced minimal NDD $\mathcal{A}_{\mathtt{x}}$ representing the set $S_{\mathtt{x}}$, with*

$$S_{\mathtt{x}} = \{(x, y) \in \mathbb{Z}^2 \mid x - y \geq 1 \wedge x - 2y \leq 2 \wedge x \geq 0 \wedge y \geq 0\}. \tag{7.6}$$

We will use this example to illustrate some definitions and theorems throughout this chapter. The sets of elements associated to the different states, i.e. $\{\langle u \rangle_{r,n} \mid u \in L_{\mathcal{A}_{\mathtt{x}}}(q_{\mathtt{I}} \to q)\}$, with $q \in \{q_1, q_2, q_3, q_4, q_5, q_6\}$ are given in Figure 7.3.

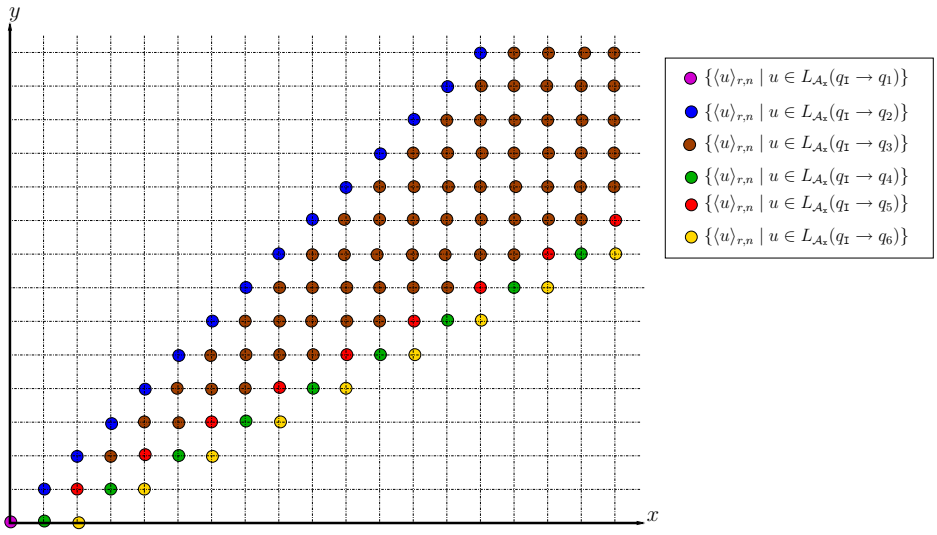


Figure 7.3: Sets associated to states in $\mathcal{A}_{\mathtt{x}}$

7.2.1 Zero-states

We show below that any simple loop in $\mathcal{A}$ labeled by a sequence of o symbols must be of size 1. We call such a loop *a zero-loop*, and states at which zero-loops are rooted are called *zero-states*. We then show that zero-states are strongly related to the characteristic cone of P. Given a zero-state $q_{\mathtt{z}}$, we characterize successively the set of words labeling paths from $q_{\mathtt{z}}$ to accepting states, i.e. $L_{\mathcal{A}}(q_{\mathtt{z}})$, and the set of words labeling loops rooted at $q_{\mathtt{z}}$, i.e. $L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$. We conclude this part with a theorem that shows an equivalence between the existence of paths between zero-states and the inclusion of languages accepted from those states.

Lemma 153. *Any simple loop in $\mathcal{A}$ labeled by a sequence of o symbols is of size* 1.

Proof. Since $\mathcal{A}$ is reduced minimal and since it represents a polyhedron, i.e. a conjunction of finitely many inequations, according to Lemma 86, $\mathcal{A}$ is permutation-free. Consequently, for any simple loop labeled by v^k for some words $v \in (\Sigma_r^n)^+$, we have $k = 1$. So, it suffices to choose $v = o$ to prove the claim. □

Example 154. *In the NDD $\mathcal{A}_{\mathtt{x}}$ of Fig.7.2, there are four zero-states, q_1, q_2, q_3 and q_5.*

We now show a relationship between the integer elements in C and those in P based on the following considerations.

- The cone C is defined as the set of vectors $\mathbf{y} \in \mathbb{Q}^n$ such that for all $\mathbf{x} \in P$, $\mathbf{x} + \mathbf{y} \in P$. We deduce that the elements of C are the vectors $\mathbf{y} \in \mathbb{Q}^n$ such that for all $\mathbf{x} \in P$ and for all $k > 0$, $\mathbf{x} + k \cdot \mathbf{y} \in P$.

- A very specific feature of the synchronous encoding scheme $E_{S(r)}$ is that suffixing an encoding u by a word v amounts to multiplying $\langle u \rangle_{r,n}$ by $r^{|v|}$ and adding $\langle ov \rangle_{r,n}$ to the resulting vector.

Lemma 155. *For all encodings $u \in (\Sigma_r^n)^*$, $\langle u \rangle_{r,n}$ belongs to $C \cap \mathbb{Z}^n$ if and only if for all elements $\langle ov \rangle_{r,n}$ of $P \cap \mathbb{Z}^n$, $\langle uv \rangle_{r,n}$ is also in $P \cap \mathbb{Z}^n$.*

Proof. Suppose that $\langle u \rangle_{r,n} \in C$. Let $v \in (\Sigma_r^n)^*$ with $\langle ov \rangle_{r,n} \in P$. We have $\mathbf{A}\langle u \rangle_{r,n} \leq 0$ and $\mathbf{A}\langle ov \rangle_{r,n} \leq \mathbf{b}$. So $\mathbf{A}(r^{|v|}\langle u \rangle_{r,n} + \langle ov \rangle_{r,n}) \leq \mathbf{b}$ and, by definition of the encoding scheme, $\mathbf{A}\langle uv \rangle_{r,n} \leq \mathbf{b}$. Therefore, $\langle uv \rangle_{r,n} \in P$.

Suppose that $\langle u \rangle_{r,n} \notin C$. We have $\mathbf{A}\langle u \rangle_{r,n} \not\leq 0$. So, there exists an inequation $\mathbf{a}.\mathbf{x} \leq b$ in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ such that $\mathbf{a}.\langle u \rangle_{r,n} > 0$. Therefore, by definition of the encoding scheme, $\mathbf{a}.\langle uo \rangle_{r,n} = r\mathbf{a}.\langle u \rangle_{r,n} > \mathbf{a}.\langle u \rangle_{r,n}$. Thanks to Lemma 85, there exists $k_{\min}$ such that for all words $w \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle uo^k w \rangle_{r,n} > b$ for all $k \geq k_{\min}$, and so, there exists a word $v \in (\Sigma_r^n)^*$ such that $\langle ov \rangle_{r,n} \in P$ and $\langle uv \rangle_{r,n} \notin P$. □

In light of the previous lemma, we expect to find the encodings of the integer elements of the characteristic cone among the prefixes of encodings labeling paths rooted at the initial state $q_{\mathtt{I}}$ and leading to accepting states in $\mathcal{A}$. The following lemma gives a very simple criterion for identifying which prefixes are the encodings of integer elements in C.

Theorem 156. *Let Q_C be the set of states q such that there is a path from q to a zero-state labeled by a sequence of o symbols.*

The integer elements of C are exactly the integer vectors whose encodings label paths from $q_{\mathtt{I}}$ to a state in Q_C.

Proof.

- Suppose $\langle u \rangle_{r,n} \in C$. From Lemma 155, for all $\langle ov \rangle_{r,n} \in P$, for all $k \in \mathbb{N}$, we have $\langle uo^k v \rangle_{r,n} \in P$, i.e. $uo^k v$ labels a path from $q_{\mathtt{I}}$ to an accepting state. Let q be the state reached via the path labeled by u from $q_{\mathtt{I}}$. So, one can follow from q a path of arbitrary length labeled by a sequence of o in $\mathcal{A}$. Since the number of states of $\mathcal{A}$ is finite, there must be a loop labeled by a sequence of o reachable from q via a path labeled by o^p for some $p \in \mathbb{N}$. From Lemma 153, the loop is a zero-loop rooted at some zero-state of $\mathcal{A}$. Therefore, by definition, $\hat{\delta}(q_{\mathtt{I}}, u) \in Q_C$.

- Let $q \in Q_C$ and $u \in (\Sigma_r^n)^+$ such that $\hat{\delta}(q_{\mathtt{I}}, u) = q$. By definition, there is a path labeled by o^p for some p, from q to a zero-state $q_{\mathtt{z}}$. So, uo^p labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$. In addition, since $\mathcal{A}$ is reduced, there is a path labeled by w from $q_{\mathtt{z}}$ to an accepting state, and the words $uo^{p+k}w$, $k \in \mathbb{N}$, label paths from $q_{\mathtt{I}}$ to an accepting state, i.e. $\langle uo^{p+k}w\rangle_{r,n} \in P$ for all $k \in \mathbb{N}$. Therefore, $\mathbf{A}\langle u\rangle_{r,n} \leq \mathbf{0}$, i.e. $\langle u\rangle_{r,n} \in C$. Otherwise, there would be an inequation $\mathbf{a}.\mathbf{x} \leq b$ from $\mathbf{Ax} \leq \mathbf{b}$ such that $\mathbf{a}.\langle u\rangle_{r,n} > 0$, and therefore, by definition of the encoding scheme, $\mathbf{a}.\langle u0\rangle_{r,n} > \mathbf{a}.\langle u\rangle_{r,n}$, and thanks to Lemma 85, there would exist $k' \in \mathbb{N}$ such that $\mathbf{a}.\langle uo^{p+k'}w\rangle_{r,n} > b$, and by definition, this would mean that $uo^{p+k'}w \notin L(\mathcal{A})$, violating the hypothesis. □

Remark 157. *Thanks to Theorem 156, given $\mathcal{A}$, we can generate in time proportional to $|\mathcal{A}|$ a deterministic NDD $\mathcal{A}_C$ accepting the encodings of the positive integer elements of C by setting all states in Q_C as the only accepting states, i.e. $\mathcal{A}_C = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_C)$. This can be done by performing a backward search, starting from all zero-states, and following only transitions labeled by o. The states reached are exactly those in Q_C.* □

We now address the characterization of the languages $L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$ and $L_{\mathcal{A}}(q_{\mathtt{z}})$ for any zero-state $q_{\mathtt{z}}$. We will show that those languages correspond to encodings of the positive integer solutions of $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ and of $\mathbf{A}'\mathbf{x} = \mathbf{0}$ respectively, for some subsystem $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ of $\mathbf{Ax} \leq \mathbf{b}$ such that for all $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $\mathbf{A}'\langle u\rangle_{r,n} = \mathbf{0}$. Intuitively, the reasoning goes as follows. For any word u labeling a path from the initial state $q_{\mathtt{I}}$ to a zero-state $q_{\mathtt{z}}$ and for any inequation $\mathbf{a}.\mathbf{x} \leq b$ in $\mathbf{Ax} \leq \mathbf{b}$, the product $\mathbf{a}.\langle u\rangle_{r,n}$ must be less or equal to 0. Otherwise, $\mathbf{a}.\langle uo^k\rangle_{r,n}$ would be arbitrarily large with increasing k, and it would not be possible to add a suffix w to uo^k such that $\mathbf{a}.\langle uo^k w\rangle_{r,n} \leq b$, i.e. there would not be any path from $q_{\mathtt{z}}$ to an accepting state, violating the hypothesis that $\mathcal{A}$ is reduced. For similar reasons, if $\mathbf{a}.\langle u\rangle_{r,n} < 0$, then, for some k, for all words w, $\mathbf{a}.\langle uo^k w\rangle_{r,n} \leq b$, i.e. the inequation $\mathbf{a}.\mathbf{x} \leq b$ does not constrain the suffixes w such that $\langle uw\rangle_{r,n} \in P$. This means that the only inequations in $\mathbf{Ax} \leq \mathbf{b}$ constraining the suffixes in a given zero-state $q_{\mathtt{z}}$ are those such that the equation $\mathbf{a}.\langle u\rangle_{r,n} = 0$ is satisfied for all words u labeling paths from the initial state to $q_{\mathtt{z}}$. We call those inequations the *pending* inequations. From above, we conclude that $L_{\mathcal{A}}(q_{\mathtt{z}})$ is the set of encodings (without the sign symbol) of the vectors in $\mathbb{N}^n$ satisfying $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ and $L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$ is the set of encodings (without the sign symbol) of the vectors in $\mathbb{N}^n$ satisfying $\mathbf{A}'\mathbf{x} = \mathbf{0}$, where $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ are the pending inequations.

Definition 158. *An inequation* $\mathbf{a}.\mathbf{x} \leq b$ *is* pending in *a zero-state* $q_{\mathtt{z}}$ *if for all*

words $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $\mathbf{a}.\langle u \rangle_{r,n} = 0$.

In order to prove that the language accepted from a zero-state $q_{\mathtt{z}}$ can be formulated in terms of the inequations of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{z}}$, we first prove two auxiliary lemmas.

We show that all words labeling paths from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$ also label loops rooted at $q_{\mathtt{z}}$. Intuitively, if the word ou labels a path from $q_{\mathtt{I}}$ to a zero-state $q_{\mathtt{z}}$, then $\mathcal{A}$ does not differentiate ou from ouo^k for all $k \in \mathbb{N}$. In other words, $\mathcal{A}$ does not distinguish the vectors $\langle ou \rangle_{r,n}$ and $r^k\langle ou \rangle_{r,n}$. Since the set represented is convex, integer vectors on the line segment between $\langle ou \rangle_{r,n}$ and $r^k\langle ou \rangle_{r,n}$ can not be distinguished by $\mathcal{A}$, and since $\langle ouu \rangle_{r,n}$ is $(r^{|u|}+1)\langle ou \rangle_{r,n}$, $\mathcal{A}$ does not differentiate $\langle ou \rangle_{r,n}$ from $\langle ouu \rangle_{r,n}$, and therefore, we expect that ou and ouu label paths leading to the same state.

Lemma 159. *For any zero-state $q_{\mathtt{z}}$ in $\mathcal{A}$, if $q_{\mathtt{z}}$ is reachable from $q_{\mathtt{I}}$ by a path labeled by ou, then there is a loop rooted at $q_{\mathtt{z}}$ labeled by u.*

Proof. Let $q_{\mathtt{z}}$ be a zero-state and $ou \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$. According to Theorem 156, $\mathbf{A}\langle ou \rangle_{r,n} \leq \mathbf{0}$.

Since $\mathcal{A}$ is minimal, by definition, there is a loop rooted at $q_{\mathtt{z}}$ labeled by u if and only if for all $w \in (\Sigma_r^n)^*$, $ouw \in L(\mathcal{A}) \Leftrightarrow ouuw \in L(\mathcal{A})$.

- Suppose that $ouw \in L(\mathcal{A})$. Then we have $A(r^{|w|}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}) \leq \mathbf{b}$, and since $\mathbf{A}\langle ou \rangle_{r,n} \leq \mathbf{0}$, we have

  $$A(r^{|w|+|u|}\langle ou \rangle_{r,n} + r^{|w|}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}) \leq A(r^{|w|}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}) \leq \mathbf{b}.$$

  By Lemma 81, $ouuw$ is an encoding of $r^{|u|+|w|}\langle ou \rangle_{r,n} + r^{|w|}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}$ and we have $ouuw \in L(\mathcal{A})$.

- Suppose that $ouuw \in L(\mathcal{A})$. Then there is a path labeled by uw from $q_{\mathtt{z}}$ to an accepting state, and therefore, since there is a zero-loop rooted at $q_{\mathtt{z}}$, for all $k \in \mathbb{N}$, $ouo^kuw \in P$ i.e. $A\langle ouo^kuw \rangle_{r,n} \leq \mathbf{b}$. Since $\mathbf{A}\langle ou \rangle_{r,n} \leq \mathbf{0}$, we have

  $$\begin{aligned} \mathbf{b} &\geq \mathbf{A}\langle ouo^kuw \rangle_{r,n} \\ &\geq \mathbf{A}\big((r^{|u|+|w|+k} + r^{|w|})\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}\big) \\ &\geq \mathbf{A}\big(r^{|u|+|w|+k+1}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}\big) \\ &\geq \mathbf{A}\langle ouo^{k+1}w \rangle_{r,n}. \end{aligned}$$

  Therefore, $ouo^{k+1}w \in L(\mathcal{A})$ and $o^{k+1}w \in L_{\mathcal{A}}(q_{\mathtt{z}})$, for all $k \in \mathbb{N}$. Since $o \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$, we conclude that $w \in L_{\mathcal{A}}(q_{\mathtt{z}})$ and $ouw \in L(\mathcal{A})$. □

Given a zero-state $q_{\mathtt{z}}$, thanks to previous lemma, concatenations of encodings labeling paths from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$ label also paths from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$. Given the definition of the pending inequations, we deduce the following lemma.

Lemma 160. *Let $q_{\mathtt{z}}$ be a zero-state. There is a word $u_{\mathtt{z}} \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that for all inequations $\mathbf{a}.\mathbf{x} \leq b$ from $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{z}}$, $\mathbf{a}.\langle u_{\mathtt{z}}\rangle_{r,n} = 0$ and for all inequations $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ not pending in $q_{\mathtt{z}}$, $\mathbf{a}.\langle u_{\mathtt{z}}\rangle_{r,n} < \min(b, -\| \mathbf{a}^+\|)$.*

Proof. Recall that $\mathbf{a}_i.\mathbf{x} \leq b_i$, $i \in \{1, \ldots, m\}$ are the inequations of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$. We partition the inequations $\mathbf{a}_i.\mathbf{x} \leq b_i$, $i \in \{1, \ldots, m\}$ into those pending in $q_{\mathtt{z}}$ and those not pending in $q_{\mathtt{z}}$. Let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.

By definition, for $i \in \{1, \ldots, m\} \setminus I_{\mathtt{p}}$, there is a word $u_i \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that $\mathbf{a}_i.\langle u_i\rangle_{r,n} \neq 0$. From Theorem 156, $\langle u_i\rangle_{r,n} \in C$ and $A\langle u_i\rangle_{r,n} \leq \mathbf{0}$. So, $\mathbf{a}_i.\langle u_i\rangle_{r,n} < 0$. Thanks to Lemma 151, $\langle ou_i\rangle_{r,n} = \langle u_i\rangle_{r,n}$ and $\hat{\delta}(q_{\mathtt{I}}, ou_i) = q_{\mathtt{z}}$. So, from Lemma 159, we deduce that $u_i \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$.

Let u be $ou_1u_2 \ldots u_m$ such that for $i \in \{1, \ldots, m\}$, $u_i \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, and $\mathbf{a}_i.\langle u_i\rangle_{r,n} < 0$ if $i \notin I_{\mathtt{p}}$. By construction, $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, and therefore, by definition, for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u\rangle_{r,n} = 0$. For all $i \in \{1, \ldots, m\} \setminus I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u\rangle_{r,n} < 0$. Indeed, by definition, $\mathbf{a}_i.\langle u_i\rangle_{r,n} < 0$, and for all $j \in \{1, \ldots, m\}$, $\mathbf{a}_i.\langle u_j\rangle_{r,n} \leq 0$. So, according to Lemma 81, we have

$$\begin{aligned}
\mathbf{a}_i.\langle u\rangle_{r,n} &= \mathbf{a}_i.(r^{|u_2\ldots u_m|}\langle u_1\rangle_{r,n} + \ldots + r^{|u_{i+1}\ldots u_m|}\langle u_i\rangle_{r,n} + \ldots + \langle u_m\rangle_{r,n}) \\
&\leq r^{|u_{i+1}\ldots u_m|}\mathbf{a}_i.\langle u_i\rangle_{r,n} \\
&< 0.
\end{aligned}$$

Finally, from Lemma 85, there exists $k \in \mathbb{N}$ such that for all $i \in \{1, \ldots, m\} \setminus I_{\mathtt{p}}$, $\mathbf{a}_i.\langle uo^k\rangle_{r,n} < \min(b_i, -\| \mathbf{a}_i{}^+\|)$ and for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle uo^k\rangle_{r,n} = 0$. Since $uo^k \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $u_{\mathtt{z}} = uo^k$ satisfies the claim. □

Since there is a word $u_{\mathtt{z}}$ such that for all inequations $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ not pending in $q_{\mathtt{z}}$, $\mathbf{a}.\langle u_{\mathtt{z}}\rangle_{r,n} \leq \min(b, -\| \mathbf{a}^+\|)$, according to the encoding scheme, for any suffix w, $\mathbf{a}.\langle u_{\mathtt{z}}w\rangle_{r,n} \leq b$. We deduce that the inequations not pending in $q_{\mathtt{z}}$ do not constrain the language accepted from $q_{\mathtt{z}}$. In addition, by definition, for all inequations $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{z}}$, $\mathbf{a}.\langle u_{\mathtt{z}}\rangle_{r,n} = 0$, and therefore, according to the encoding scheme, all suffixes w labeling a path from $q_{\mathtt{z}}$ to an accepting state have to satisfy $\mathbf{a}.\langle ow\rangle_{r,n} \leq b$. Thanks to this result, we can specify $L_{\mathcal{A}}(q_{\mathtt{z}})$.

Theorem 161. *Let $q_{\mathtt{z}}$ be a zero-state and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.*

$$L_{\mathcal{A}}(q_{\mathtt{z}}) = \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i \langle ow \rangle_{r,n} \leq b_i\}.$$

Proof. From Lemma 160, there is a word $u_{\mathtt{z}} \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{z}})$ such that for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i \langle u_{\mathtt{z}} \rangle_{r,n} = 0$ and for all $i \in \{1, \ldots, m\} \backslash I_{\mathtt{p}}$, $\mathbf{a}_i \langle u_{\mathtt{z}} \rangle_{r,n} < \min(b_i, -\| \mathbf{a}_i{}^+ \|)$.

- Suppose that $\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i \langle ow \rangle_{r,n} \leq b_i$. According to Lemma 82, for all $i \in \{1, \ldots, m\} \setminus I_{\mathtt{p}}$, we have $\mathbf{a}_i.\langle u_{\mathtt{z}} w \rangle_{r,n} < \min(b_i, -\| \mathbf{a}_i{}^+ \|)$. In addition, according to Lemma 81, for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u_{\mathtt{z}} w \rangle_{r,n} = \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i$. Therefore, $\mathbf{A} \langle u_{\mathtt{z}} w \rangle_{r,n} \leq \mathbf{b}$, i.e. $w \in L_{\mathcal{A}}(q_{\mathtt{z}})$.

- Suppose that $w \in L_{\mathcal{A}}(q_{\mathtt{z}})$. Then, by definition, $u_{\mathtt{z}} w \in L(\mathcal{A})$ and $\mathbf{A} \langle u_{\mathtt{z}} w \rangle_{r,n} \leq \mathbf{b}$. Since for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u_{\mathtt{z}} \rangle_{r,n} = 0$, we have

  $$\mathbf{a}_i.\langle u_{\mathtt{z}} w \rangle_{r,n} = \mathbf{a}_i.\langle ow \rangle_{r,n}.$$

  We conclude that for all $i \in I_{\mathtt{p}}$, we have

  $$\mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i.$$

□

The next theorem is also a consequence of the fact that only pending inequations play a role in the language accepted from a zero-state.

Theorem 162. *Let $q_{\mathtt{z}}$ be a zero-state and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.*

$$L_{\mathcal{A}}(q_{\mathtt{z}} \rightarrow q_{\mathtt{z}}) = \{u \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ou \rangle_{r,n} = 0\}.$$

Proof.

- Suppose $\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ou \rangle_{r,n} = 0$. Thanks to Lemma 81, for all $i \in I_{\mathtt{p}}$ and for all $w \in (\Sigma_r^n)^*$, we have

  $$\mathbf{a}_i.\langle uw \rangle_{r,n} = \mathbf{a}_i.\langle ow \rangle_{r,n}. \tag{7.7}$$

According to Theorem 161, we have $L_{\mathcal{A}}(q_{\mathtt{z}}) = \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i\}$, and therefore, by definition, we get

$$u \div L_{\mathcal{A}}(q_{\mathtt{z}}) = \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle uw \rangle_{r,n} \leq b_i\}. \tag{7.8}$$

From (7.7) and (7.8), we deduce that

$$\begin{aligned} u \div L_{\mathcal{A}}(q_{\mathtt{z}}) &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle uw \rangle_{r,n} \leq b_i\} \\ &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i\} \\ &= L_{\mathcal{A}}(q_{\mathtt{z}}). \end{aligned}$$

Let $q' = \hat{\delta}(q_{\mathtt{z}}, u)$. By definition, $L_{\mathcal{A}}(q') = L_{\mathcal{A}}(q)$. Therefore, since $\mathcal{A}$ is reduced minimal, $q' = q_{\mathtt{z}}$, i.e. $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$.

- Suppose $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$. Let $v \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$. By hypothesis, $vu \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$. Therefore, by definition, for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle v \rangle_{r,n} = 0$ and $\mathbf{a}_i.\langle vu \rangle_{r,n} = 0$, which implies that $\mathbf{a}_i.\langle ou \rangle_{r,n} = 0$ given Lemma 81. □

In the following theorem, we show that the language accepted from a zero-state q_1 is included in the language accepted from another zero-state q_2 if and only if there is a path from q_1 to q_2, which occurs if and only if any word labeling a loop rooted at q_1 labels a loop rooted at q_2. This result is a direct consequence of the fact that the inequations of $\mathbf{Ax} \leq \mathbf{b}$ pending in q_2 form a subset of the inequations pending in q_1 and that the language accepted from a zero-state $q_{\mathtt{z}}$ is expressed in terms of inequations of $\mathbf{Ax} \leq \mathbf{b}$ pending in $q_{\mathtt{z}}$.

Theorem 163. *Let q_1 and q_2 be zero-states. The following assertions are equivalent :*

1. *There exists a path from q_1 to q_2.*
2. $L_{\mathcal{A}}(q_1 \to q_1) \subseteq L_{\mathcal{A}}(q_2 \to q_2)$.
3. $L_{\mathcal{A}}(q_1) \subseteq L_{\mathcal{A}}(q_2)$.

Proof. We will prove that (1) ⇒ (2) ⇒ (3) ⇒ (1).

Let $I_1 \subseteq \{1, \ldots, m\}$ be such that $i \in I_1$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in q_1, and similarly, let $I_2 \subseteq \{1, \ldots, m\}$ be such that $i \in I_1$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in q_2.

From Theorem 161, we have

$$L_{\mathcal{A}}(q_1) = \{w \mid \bigwedge_{i\in I_1} \mathbf{a}_i.\langle ow\rangle_{r,n} \leq b_i\} \tag{7.9}$$

$$L_{\mathcal{A}}(q_2) = \{w \mid \bigwedge_{i\in I_2} \mathbf{a}_i.\langle ow\rangle_{r,n} \leq b_i\}, \tag{7.10}$$

and from Theorem 162, we have

$$L_{\mathcal{A}}(q_1 \to q_1) = \{u \mid \bigwedge_{i\in I_1} \mathbf{a}_i.\langle ou\rangle_{r,n} = 0\} \tag{7.11}$$

$$L_{\mathcal{A}}(q_2 \to q_2) = \{u \mid \bigwedge_{i\in I_2} \mathbf{a}_i.\langle ou\rangle_{r,n} = 0\}. \tag{7.12}$$

Finally, recall that thanks to Lemma 151, for any state $q \in Q$ and word $u \in (\Sigma_r^n)^+$, if $\hat{\delta}(q_{\mathtt{I}}, u) = q$, then $\hat{\delta}(q_{\mathtt{I}}, ou) = q$.

- Suppose that there is a path from q_1 to q_2 labeled by w. From Lemma 160, there is a word $u_1 \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_1)$ such that

$$\begin{aligned} \mathbf{a}_i.\langle u_1\rangle_{r,n} &= 0 \quad \text{if} \quad i \in I_1 \\ \mathbf{a}_i.\langle u_1\rangle_{r,n} &< \min(b_i, -\|\mathbf{a}_i^+\|) \quad \text{if} \quad i \in \{1,\ldots,m\} \setminus I_1. \end{aligned}$$

  By hypothesis, $u_1 w$ labels a path from $q_{\mathtt{I}}$ to q_2 and, thanks to Lemma 82, we deduce that for all $i \in \{1,\ldots,m\} \setminus I_1$, $\mathbf{a}_i.\langle u_1 w\rangle_{r,n} < \min(b_i, -\|\mathbf{a}_i^+\|) < 0$. Hence, by definition, $I_1 \supseteq I_2$. From (7.11) and (7.12), we conclude that $L_{\mathcal{A}}(q_1 \to q_1) \subseteq L_{\mathcal{A}}(q_2 \to q_2)$.

- Suppose that $L_{\mathcal{A}}(q_1 \to q_1) \subseteq L_{\mathcal{A}}(q_2 \to q_2)$. Let $u_1 \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_1)$ and let $u_2 \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_2)$. Thanks to Theorem 156, we have $\mathbf{A}\langle u_1\rangle_{r,n} \leq \mathbf{0}$ and $\mathbf{A}\langle u_2\rangle_{r,n} \leq \mathbf{0}$. Let $w \in L_{\mathcal{A}}(q_1)$. By definition, $u_1 w$ is in $\mathcal{A}$ and therefore, $ou_1 w$ belongs also to $L(\mathcal{A})$. So, by hypothesis, we have

$$\mathbf{A}(r^{|w|}\langle ou_1\rangle_{r,n} + \langle ow\rangle_{r,n}) \leq \mathbf{b}. \tag{7.13}$$

  Since $\mathbf{A}\langle u_2\rangle_{r,n} \leq \mathbf{0}$, we deduce that

$$\mathbf{A}(r^{|u_1|+|u_2|+|w|}\langle u_2\rangle_{r,n} + r^{|w|}\langle ou_1\rangle_{r,n} + \langle ow\rangle_{r,n}) \leq \mathbf{b}. \tag{7.14}$$

  From (7.14) and by defintion of the encoding scheme, we deduce that $\mathbf{A}\langle u_2 u_1 w\rangle_{r,n} \leq \mathbf{b}$. Thus, by hypothesis, the word $u_2 u_1 w$ is in $L(\mathcal{A})$ and we have

$$u_1 w \in L_{\mathcal{A}}(q_2). \tag{7.15}$$

From Lemma 159, u_1 labels a loop rooted at q_1, and therefore, by hypothesis, u_1 labels a loop rooted at q_2. So, since $u_1 w$ is accepted from q_2 and since $\mathcal{A}$ is deterministic, w is accepted from q_2. We conclude that $L_{\mathcal{A}}(q_1) \subseteq L_{\mathcal{A}}(q_2)$.

- Suppose $L_{\mathcal{A}}(q_1) \subseteq L_{\mathcal{A}}(q_2)$. Let $u_1 \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_1)$ and let $u_2 \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_2)$, and so $\hat{\delta}(q_{\mathtt{I}}, ou_2) = q_2$. Thanks to Lemma 48, we have

$$u_1 \div L(\mathcal{A}) \subseteq u_2 \div L(\mathcal{A}). \tag{7.16}$$

Therefore, by definition, we have

$$u_1 u_2 \div L(\mathcal{A}) \subseteq u_2 u_2 \div L(\mathcal{A}). \tag{7.17}$$

From Lemma 159, there is a loop rooted at q_2 labeled by u_2. So, thanks to Lemma 48, $u_2 u_2 \div L(\mathcal{A}) = u_2 \div L(\mathcal{A})$, and we deduce from (7.17) that

$$u_1 u_2 \div L(\mathcal{A}) \subseteq u_2 \div L(\mathcal{A}). \tag{7.18}$$

In addition, from Theorem 156, $\langle u_1 \rangle_{r,n} \in C$ and from Lemma 155, for all word $w \in (\Sigma_r^n)^*$, we have

$$ou_2 w \in L(\mathcal{A}) \Rightarrow u_1 u_2 w \in L(\mathcal{A}). \tag{7.19}$$

So, we have

$$u_1 u_2 \div L(\mathcal{A}) \supseteq ou_2 \div L(\mathcal{A}) = u_2 \div L(\mathcal{A}). \tag{7.20}$$

From (7.18) and (7.20), we deduce that

$$u_1 u_2 \div L(\mathcal{A}) = u_2 \div L(\mathcal{A}). \tag{7.21}$$

Since $\mathcal{A}$ is reduced minimal, $\hat{\delta}(q_1, u_2) = \hat{\delta}(q_{\mathtt{I}}, u_1 u_2) = \hat{\delta}(q_{\mathtt{I}}, u_2) = q_2$. We conclude that there is a path from q_1 to q_2 labeled by u_2. □

7.2.2 Zero-SCCs

In this subsection, we characterize the SCCs having a zero-state. We show that there is at most one zero-state in any SCC and we partition the maximal SCCs of $\mathcal{A}$ between those having a zero-state (called *zero-SCCs*) and the other SCCs. In the previous section, we have shown that the words labeling loops rooted at

some zero-state $q_{\mathtt{z}}$ are the encodings (without sign symbol) of the elements in the intersection between a $\mathbb{Q}$-vector space V and $\mathbb{N}^n$. We will show in this section that the languages corresponding to words labeling paths from $q_{\mathtt{z}}$ to another state in the same SCC are the encodings of the intersection between a coset of V and $\mathbb{N}^n$. Therefore, the particular $\mathbb{Q}$-vector space V characterizes the zero-SCC, and in particular, we define the *dimension* of a zero-SCC as the dimension of $V \cap \mathbb{N}^n$. Finally, we characterize the labels of the incoming transitions in states of a zero-SCC as well as the number of incoming transitions.

Lemma 164. *In any maximal SCC of $\mathcal{A}$, there is at most one zero-state.*

Proof. Let q_1 and q_2 be zero-states in a maximal SCC $\mathcal{S}$. From Theorem 163, $L_{\mathcal{A}}(q_1) = L_{\mathcal{A}}(q_2)$ and since $\mathcal{A}$ is reduced minimal, $q_1 = q_2$. □

Definition 165. *A* zero-SCC *is a maximal strongly connected component having a zero-state.*

From Lemma 164, there is exactly one zero-state in any zero-SCC. Note that some SCCs do not have any zero-state.

According to Theorem 163, the words labeling loops rooted at the zero-state of a zero-SCC are the encodings (from which the sign symbol o has been removed) of the natural solutions of a system of homogeneous equations, i.e. the elements in the intersection of a vector space with $\mathbb{N}^n$. We define the *dimension of* a zero-SCC as the dimension of this intersection.

Definition 166. *Let $\mathcal{S}$ be a zero-SCC and let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$. The* dimension *of $\mathcal{S}$, written* $\dim(\mathcal{S})$*, is the dimension of the set* $\{\langle ou\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \rightarrow q_{\mathtt{z}})\}$.

Example 167. *In the NDD $\mathcal{A}_{\mathtt{x}}$ of Fig.7.2, there are four zero-SCCs. The dimension of the zero-SCC containing the zero-state q_1 (resp. q_2, q_3, q_5) is 0 (resp. 1, 2 and 1). Note that q_6 forms a SCC with no zero-state.*

By definition, for NDDs representing the positive integer elements of polyhedra in $\mathbb{Q}^n$, the dimensions of the zero-SCCs are at least 0 and at most n. The following theorems describe the structure of the zero-SCCs in these extreme cases. We show that in both cases, there is only one state in the SCC, the zero-state. In addition, if $\dim(\mathcal{S}) = n$, then for all $\alpha \in \Sigma_r^n$, there is a simple loop labeled by α rooted at the zero-state, and if $\dim(\mathcal{S}) = 0$, then there is only one simple loop rooted at $q_{\mathtt{z}}$ and it is labeled by o.

Theorem 168. *Let $\mathcal{S}$ be a zero-SCC and let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$.*
If $\dim(\mathcal{S}) = n$, then $L_{\mathcal{A}}(q_{\mathtt{z}}) = L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = (\Sigma_r^n)^*$.

Proof. Let $S = \{\langle ou \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}$, and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.

According to Theorem 162, $L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = \{u \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ou \rangle_{r,n} = 0\}$, and therefore, $S = V \cap \mathbb{N}^n$ for some $\mathbb{Q}$-vector space V. If $\dim(\mathcal{S}) = n$, then by definition, there exist n vectors linearly independent in V and $V = \mathbb{Z}^n$. Therefore, $S = \mathbb{N}^n$, and for all $\alpha \in \Sigma_r^n$, $\delta(q_{\mathtt{z}}, \alpha) = q_{\mathtt{z}}$. Since $\mathcal{A}$ is reduced minimal and since $q_{\mathtt{z}}$ is the only reachable state from $q_{\mathtt{z}}$, $q_{\mathtt{z}}$ is an accepting state and $q_{\mathtt{z}} \in Q_{\mathtt{F}}$. □

Theorem 169. *Let $\mathcal{S}$ be a zero-SCC and let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$.*
If $\dim(\mathcal{S}) = 0$, then there is only one simple loop rooted at $q_{\mathtt{z}}$, and this loop is labeled by o.

Proof. This is a direct consequence of the definition of the dimension of a zero-SCC and of the fact that the zero-loops are of size 1. □

Example 170. *In Fig.7.2, the NDD $\mathcal{A}_{\mathtt{x}}$ representing $S_{\mathtt{x}} \subseteq \mathbb{Z}^2$ is such that the dimension of the zero-SCC containing the zero-state q_1 is 0 and the dimension of the zero-SCC containing the zero-state q_3 is 2.*

Since the set of words labeling loops rooted at a zero-state is defined by the homogeneous system of equations corresponding to the pending inequations, the set of words labeling paths from the zero-state $q_{\mathtt{z}}$ to a state q in the same zero-SCC corresponds to the same linear system, but whose right-hand side vector is possibly different than 0, i.e. if the words labeling loops rooted at the zero-state $q_{\mathtt{z}}$ are the encodings (without sign symbol) of the positive integer solutions of $\mathbf{A}'\mathbf{x} = \mathbf{0}$, then the words labeling paths from $q_{\mathtt{z}}$ to a state q in the same zero-SCC are the encodings (without sign symbol) of the positive integer solutions of $\mathbf{A}'\mathbf{x} = \mathbf{b}'$ for some $\mathbf{b}'$. Indeed, on the one hand, if for an equation $\mathbf{a}.\mathbf{x} = 0$ from $\mathbf{A}'\mathbf{x} = \mathbf{0}$ there are two words u and v labeling paths from $q_{\mathtt{z}}$ to q such that $\mathbf{a}.\langle ou \rangle_{r,n} \neq \mathbf{a}.\langle ov \rangle_{r,n}$, then either $\mathbf{a}.\langle ouw \rangle_{r,n} \neq 0$ or $\mathbf{a}.\langle ovw \rangle_{r,n} \neq 0$ for some w with $uw, vw \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$. On the other hand, since the pending inequations specify the suffixes leading to accepting states, if two vectors are not distinguishable with respect to the pending inequations, their encodings should label paths to the same state in the NDD. Formally, we have the following theorem.

Theorem 171. *Let $\mathcal{S}$ be a zero-SCC, let $q_{\mathtt{z}}$ be its zero-state and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.*

For all states q in $\mathcal{S}$ and for all $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$,

$$L_{\mathcal{A}}(q_{\mathtt{z}} \to q) = \{u \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(\langle ou \rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q} \rangle_{r,n}) = 0\}.$$

Proof. Let $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$ and $v_{qq_{\mathtt{z}}} \in L_{\mathcal{A}}(q \to q_{\mathtt{z}})$.

- Suppose $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. By definition, we the word $uv_{qq_{\mathtt{z}}}$ labels a loop rooted at $q_{\mathtt{z}}$. Thanks to Theorem 162, we have

  $$L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ow \rangle_{r,n} = 0\}.$$

  Therefore, we deduce that the following assertions hold.

  $$\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q} v_{qq_{\mathtt{z}}} \rangle_{r,n} = 0\,,$$

  $$\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ouv_{qq_{\mathtt{z}}} \rangle_{r,n} = 0.$$

  So, for all $i \in I_{\mathtt{p}}$, according to Lemma 81, we have

  $$\begin{aligned} \mathbf{a}_i.(r^{|v_{qq_{\mathtt{z}}}|}\langle ov_{q_{\mathtt{z}}q} \rangle_{r,n} + \langle ov_{qq_{\mathtt{z}}} \rangle_{r,n}) &= \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q} v_{qq_{\mathtt{z}}} \rangle_{r,n}, \\ &= \mathbf{a}_i.\langle ouv_{qq_{\mathtt{z}}} \rangle_{r,n}, \\ &= \mathbf{a}_i.(r^{|v_{qq_{\mathtt{z}}}|}\langle ou \rangle_{r,n} + \langle ov_{qq_{\mathtt{z}}} \rangle_{r,n}). \end{aligned}$$

  We conclude that for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle ou \rangle_{r,n} = \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q} \rangle_{r,n}$.

- Suppose $\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(\langle ou \rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q} \rangle_{r,n}) = 0$. Thanks to Lemma 48, we have $L_{\mathcal{A}}(q) = v_{q_{\mathtt{z}}q} \div L_{\mathcal{A}}(q_{\mathtt{z}})$, and thanks to Theorem 161, we have

  $$L_{\mathcal{A}}(q_{\mathtt{z}}) = \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i\}.$$

  Therefore, we have

  $$\begin{aligned} u \div L_{\mathcal{A}}(q_{\mathtt{z}}) &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ouw \rangle_{r,n} \leq b_i\} \\ &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(r^{|w|}\langle ou \rangle_{r,n} + \langle ow \rangle_{r,n}) \leq b_i\} \\ &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(r^{|w|}\langle ov_{q_{\mathtt{z}}q} \rangle_{r,n} + \langle ow \rangle_{r,n}) \leq b_i\} \\ &= \{w \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q} w \rangle_{r,n} \leq b_i\} \\ &= v_{\mathtt{z}q} \div L_{\mathcal{A}}(q_{\mathtt{z}}). \end{aligned}$$

Since $\mathcal{A}$ is reduced minimal, thanks to Proposition 53, we conclude that $\hat{\delta}(q_{\mathtt{z}}, u) = \hat{\delta}(q_{\mathtt{z}}, u_{q_{\mathtt{z}}q}) = q$ and $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. □

The following theorem characterizes further the languages corresponding to paths from a zero-state to other states in the same zero-SCC.

Theorem 172. *Let $\mathcal{S}$ be a zero-SCC, let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$ and let $d = \dim(\mathcal{S})$.*

There exists an integer matrix $\mathbf{B}$ of rank $n - d$ such that for all states $q \in \mathcal{S}$ and words $v_q \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$,

- $\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\} = \{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_q\rangle_{r,n}) = \mathbf{0}\}$,
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_q\rangle_{r,n}) = \mathbf{0}\}$.

Proof. If $d = n$, then from Theorem 168, $\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\} = \mathbb{N}^n$ and $q_{\mathtt{z}}$ is the only state in $\mathcal{S}$. Therefore, one can chose $\mathbf{B}$ to be the matrix with one row and n columns whose elements are all 0.

In the following, we assume that $d < n$. Let $S = \{\langle ou\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}$ and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.

From Theorem 162,

$$S = \{\langle ou\rangle_{r,n} \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ou\rangle_{r,n} = 0\}. \tag{7.22}$$

Since $\dim(S) = d$, there exist d words $u_1, \ldots, u_d \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$ and d vectors $\mathbf{x}_1, \ldots, \mathbf{x}_d$ linearly independent in S with $\langle ou_i\rangle_{r,n} = \mathbf{x}_i$, $i \in \{1, \ldots, d\}$. So, there exists $\mathbf{B} \in \mathbb{Z}^{(n-d)\times n}$ with $\mathrm{rank}(\mathbf{B}) = n - d$ such that $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(\{\mathbf{x}_1, \ldots, \mathbf{x}_d\})$ and $S = \{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$.

Let q be a state of $\mathcal{S}$ and let $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. Since $q_{\mathtt{z}}$ and q are in the same SCC, there exists a word $v_{qq_{\mathtt{z}}}$ labeling a path from q to $q_{\mathtt{z}}$.

- Suppose $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. Then we have $uv_{qq_{\mathtt{z}}} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})$ and by definition of $\mathbf{B}$, we deduce that

  $$\mathbf{B}\langle ouv_{qq_{\mathtt{z}}}\rangle_{r,n} = \mathbf{0} = \mathbf{B}\langle ov_{q_{\mathtt{z}}q}v_{qq_{\mathtt{z}}}\rangle_{r,n}.$$

  Therefore, we have $\mathbf{B}(\langle ou\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$.

- Suppose $\mathbf{x} \in \mathbb{N}^n$ satisfies $\mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$. By definition, we have

  $$\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} = \sum_{j=1}^{d} a_j\mathbf{x}_j,$$

for some $a_1, \ldots, a_d \in \mathbb{Q}$.

Therefore, since $\mathbf{x}_j \in S$ for $j \in \{1, \ldots, d\}$, we deduce that for all $i \in I_{\mathtt{p}}$, we have $\mathbf{a}_i.(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = 0$. Since $\mathbf{x} \in \mathbb{N}^n$, there exists a word $u \in (\Sigma_r^n)^*$ with $\langle ou\rangle_{r,n} = \mathbf{x}$, and for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.(\langle ou\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = 0$. From Theorem 171, we have

$$L_{\mathcal{A}}(q_{\mathtt{z}} \to q) = \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(\langle ow\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = 0\}.$$

Therefore, we conclude that $u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$.

We have therefore proved that

$$\{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n})\} = \{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}. \tag{7.23}$$

By definition of the affine hull, we have

$$\mathrm{aff}_{\mathbb{Q}}(\{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\} \tag{7.24}$$

From (7.23) and (7.24), we deduce that

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\} \supseteq \mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) \tag{7.25}$$

Since $\mathrm{rank}(\mathbf{B}) = n - d$, we have

$$\dim(\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}) = d. \tag{7.26}$$

In addition, by definition, $u_1v_{q_{\mathtt{z}}q}$, …, $u_dv_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$, and $\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}$, $\langle ou_1v_{q_{\mathtt{z}}q}\rangle_{r,n}$, …, $\langle ou_dv_{q_{\mathtt{z}}q}\rangle_{r,n}$ are affinely independent, and

$$\dim(\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) \geq d. \tag{7.27}$$

From (7.25), (7.26) and (7.27), we conclude that

$$\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}. \tag{7.28}$$

□

Based on the previous theorem, we introduce the notion of representative matrix.

Definition 173. *A* representative matrix *of a zero-SCC $\mathcal{S}$ of dimension d is an integer matrix $\mathbf{B}$ of rank $n - d$ such that if $q_{\mathtt{z}}$ is the zero-state of $\mathcal{S}$, for all states $q \in \mathcal{S}$ and all $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$,*

- $\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\} = \{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}$,
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}$.

Corollary 174. *For each zero-SCC $\mathcal{S}$, there exists a representative matrix.*

Proof. Direct consequence of Theorem 172 and of the definition of a representative matrix. □

As it will become clear in the remaining of this section, the notions of zero-SCC, and in particular of representative matrix, will play a key role in our algorithm generating a formula for the characteristic cone of P. In the next theorem, we show that one can efficiently compute a representative matrix.

Theorem 175. *Let $\mathcal{S}$ be a zero-SCC, let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$ and let $Q_{\mathcal{S}} \subseteq Q$ be the set of states in $\mathcal{S}$.*

There exists an algorithm GETREPRESENTATIVEMATRIX *which, given the reduced minimal NDD $\mathcal{A}$ and the zero-SCC $\mathcal{S}$, computes a representative matrix of $\mathcal{S}$ and whose time complexity is $\mathcal{O}(|Q_{\mathcal{S}}| \cdot |\Sigma_r^n| \cdot n^2)$.*

Proof. It suffices to create a new NDD $\mathcal{A}_{\mathcal{S}} = (Q'_{\mathcal{S}}, \Sigma_r^n, \delta', q'_{\mathtt{I}}, \{q_{\mathtt{z}}\})$, such that the set of states $Q'_{\mathcal{S}}$ and the transitions correspond to the set of states and the transitions in $\mathcal{S}$, except for a new state $q'_{\mathtt{I}}$ which is set as the initial state of $\mathcal{A}_{\mathcal{S}}$ and for a new transition, from $q'_{\mathtt{I}}$ to $q_{\mathtt{z}}$ labeled by o. Finally, $q_{\mathtt{z}}$ is marked as the only accepting state in $\mathcal{A}_{\mathcal{S}}$. By construction, the set represented by $\mathcal{A}_{\mathcal{S}}$ is $S_{\mathcal{S}} = \{\langle ou\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}$. Therefore, given the NDD $\mathcal{A}_{\mathcal{S}}$ as input, the output of algorithm QAFFINEHULLEQUATIONS is a system of linear equations $\mathbf{Bx} = \mathbf{0}$ such that $\mathbf{B}$ is a representative matrix of $\mathcal{S}$. By construction and thanks to Theorem 120, the time complexity for generating a representative matrix of $\mathcal{S}$ is $\mathcal{O}(|Q_{\mathcal{S}}| \cdot |\Sigma_r^n| \cdot n^2)$. □

We conclude this section by characterizing the incoming transitions in states of a zero-SCC.

Lemma 176. *For each α in Σ_r^n and each state q of a zero-SCC $\mathcal{S}$ of $\mathcal{A}$, there is at most one state $q' \in \mathcal{S}$ such that $\delta(q', \alpha) = q$.*

Proof. Let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$, let $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$ and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$.

From Theorem 171,

$$L_{\mathcal{A}}(q_{\mathtt{z}} \to q) = \{v \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.(\langle ov\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = 0\}$$

Therefore, for all $u_1, u_2 \in (\Sigma_r^n)^*$ such that $u_1\alpha, u_2\alpha \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$, for all $i \in I_{\mathtt{p}}$,

$$\begin{aligned}
\mathbf{a}_i.\langle ou_1\alpha\rangle_{r,n} &= \mathbf{a}_i.(r\langle ou_1\rangle_{r,n} + \langle o\alpha\rangle_{r,n}) \\
&= \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} \\
\mathbf{a}_i.\langle ou_2\alpha\rangle_{r,n} &= \mathbf{a}_i.(r\langle ou_2\rangle_{r,n} + \langle o\alpha\rangle_{r,n}) \\
&= \mathbf{a}_i.\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}.
\end{aligned}$$

Therefore, $\forall i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle ou_1\rangle_{r,n} = \mathbf{a}_i.\langle ou_2\rangle_{r,n}$. Considering Theorem 171, we conclude that $\hat{\delta}(q_{\mathtt{z}}, u_1) = \hat{\delta}(q_{\mathtt{z}}, u_2)$, i.e. all transitions incoming in q from states in $\mathcal{S}$ and labeled by α originate from the same state, i.e. there is at most one incoming transition originating from a state in $\mathcal{S}$ labeled by α. □

Theorem 177. *Let $\mathcal{S}$ be a zero-SCC and let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$ and let $d = \dim(\mathcal{S})$.*

Each state of $\mathcal{S}$ has r^d incoming transitions from states in $\mathcal{S}$.

Proof. See Section 7.8.1. □

7.2.3 Zero-SCCs and Faces of the Characteristic Cone

In this section, we emphasize the relationship between zero-SCCs of the NDD $\mathcal{A}$ representing $P \cap \mathbb{N}^n$ and the faces of the characteristic cone C of P. Recall first that one can associate a vector space V to each zero-SCC such that V is equal to the linear hull of the vectors whose encodings (without the o sign-symbol) label loops rooted at the zero-state of the zero-SCC. We present two kinds of associations between faces and zero-SCCs. First we show that for any zero-SCC $\mathcal{S}$, the vector space of $\mathcal{S}$ is the linear hull of one and only one face of C. Second, we show that for each face F of C, there is one and only one zero-SCC $\mathcal{S}$ such that there is an encoding of an element of F labeling a path from the initial state to the zero-state of $\mathcal{S}$ and the linear hull of F is included in the vector space of $\mathcal{S}$. Then, based on the properties of the facets, we give some technical results which form the justifications of the algorithm given in the next section detailing how one can generate efficiently a formula whose set of solutions is C.

Recall that C is the characteristic cone of $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$. Therefore, by definition, $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0}\}$. Also, since $P \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$, we have $C \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$. Finally, we have defined the matrix $\mathbf{C}$ such that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$, and no inequations in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ is redundant in $\mathbf{C}\mathbf{x} \leq \mathbf{0}$.

Since $C \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$, we have the following lemma.

Lemma 178. *For each face F of C, $\lin_{\mathbb{Q}}(F) = \lin_{\mathbb{Q}}\big(\lin_{\mathbb{Q}}(F) \cap \mathbb{N}^n\big)$.*

Proof. See Section 7.8.2. □

Let $q_{\mathtt{z}}$ be a zero-state and let $F = \{\mathbf{x} \in C \mid \mathbf{A}'\mathbf{x} = \mathbf{0}\}$, where $\mathbf{A}'\mathbf{x} \leq \mathbf{b}'$ are the inequations of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{z}}$. The set F is by definition a face of C. Thanks to Theorem 156 and by definition of the pending inequations, the vectors whose encodings label paths from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$ are in F. In addition, by definition, the linear hull over $\mathbb{Q}$ of F is the set of solution to the system of linear equations corresponding to the pending equations. This gives the possibility of expressing Lemma 160 and Theorem 162 in terms of the face F. The following lemma incorporate those considerations.

Lemma 179. *Let $q_{\mathtt{z}}$ be a zero-state and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$. Let $F = \{\mathbf{x} \in C \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}$.*

- *For all $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $\langle u \rangle_{r,n} \in F$,*
- *There exists a word $u \in (\Sigma_r^n)^*$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that $\langle u \rangle_{r,n} \in F$ and for all proper faces F' of F, $\langle u \rangle_{r,n} \notin F'$,*
- *$L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = \{v \in (\Sigma_r^n)^* \mid \langle ov \rangle_{r,n} \in \lin_{\mathbb{Q}}(F)\}$,*
- *$\lin_{\mathbb{Q}}(\{\langle ov \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}) = \lin_{\mathbb{Q}}(F)$*

Proof. See Section 7.8.3. □

Thanks to the previous lemma, we deduce that the following theorem, and this constitutes the first association between zero-SCCs and faces of C.

Theorem 180. *For each zero-SCC $\mathcal{S}$, one can associate one and only one face $F_{\mathcal{S}}$ of C such that*

$$\lin_{\mathbb{Q}}(F_{\mathcal{S}}) = \lin_{\mathbb{Q}}(\{\langle ou \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}),$$

where $q_{\mathtt{z}}$ is the zero-state of $\mathcal{S}$.

Proof. This is a direct consequence of Lemma 179 and of the fact that for all faces F_1, $_2$ of a cone, $\lin_{\mathbb{Q}}(F_1) = \lin_{\mathbb{Q}}(F_2)$ iff $F_1 = F_2$, as proved in Lemma 30. □

Remark 181. *Given a zero-SCC $\mathcal{S}$ and a representative matrix $\mathbf{B}$ of $\mathcal{S}$, if $F_{\mathcal{S}}$ is such that $\lin_{\mathbb{Q}}(F) = \lin_{\mathbb{Q}}(\{\langle ou \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\})$ where $q_{\mathtt{z}}$ is the zero-state of $\mathcal{S}$, then, by definition of a representative matrix*

$$\lin_{\mathbb{Q}}(F_{\mathcal{S}}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}.$$

□

We now introduce the second association between faces and zero-SCCs. We first show in the following lemma how each face can be related to one unique zero-state. The idea is that a face F of C is the set of solutions of a system of homogeneous inequations corresponding to the inequations $\mathbf{a}.\mathbf{x} \leq b$ defining P. Among those inequations, one distinguishes the *implicit equations* $\mathbf{a}.\mathbf{x} \leq 0$ such that the corresponding equation $\mathbf{a}.\mathbf{x} = 0$ is satisfied by all elements in F from the other inequations. An element of F which does not belong to a proper face satisfies strictly all inequations which are not implicit equations. By definition of the encoding scheme, we deduce that suffixing any encoding u of such (integer) element by o symbols leaves the right-hand sides of the implicit equations unchanged i.e.

$$0 = \mathbf{a}.\langle u\rangle_{r,n} = \mathbf{a}.\langle uo\rangle_{r,n} = \mathbf{a}.\langle uoo\rangle_{r,n} = \dots$$

but the right-hand-sides of the inequations which are not implicit are decreasing, i.e.

$$\mathbf{a}.\langle u\rangle_{r,n} > \mathbf{a}.\langle uo\rangle_{r,n} > \mathbf{a}.\langle uoo\rangle_{r,n} > \dots$$

At some point, $\mathbf{a}.\langle uo^k w\rangle_{r,n} \leq b$ for all suffixes w, and the inequations are no longer constraining. So, for sufficiently large k, for all encodings u, v of integer elements of F not in any proper face of F, the NDD does not differentiate uo^k and vo^k, and so uo^k and vo^k label paths from $q_{\mathtt{I}}$ to the same state.

Lemma 182. *For each face F of the characteristic cone C, there exists one and only one zero-state, denoted q_F, such that the encodings, possibly suffixed by a sequence of o symbols, of all integer elements in F which do not belong to a proper face of F label paths from $q_{\mathtt{I}}$ to q_F.*

In addition, the state q_F is such that

- $L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle ov\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)\}$, *and*
- $\mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) \supseteq \mathrm{lin}_{\mathbb{Q}}(F)$.

Proof. See Section 7.8.4. □

Theorem 183. *For each face F of C, there is one and only one zero-SCC $\mathcal{S}_F$ such that if $q_{\mathtt{z}}$ is the zero-state of $\mathcal{S}_F$, the following assertions hold.*

- *There is at least one encoding of an element of F which labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$, and*
- $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \mathrm{lin}_{\mathbb{Q}}(\{\langle ou\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\})$.

Proof. Thanks to Lemma 182, there is at least one zero-SCC for which the assertions hold. So, it suffices to show that this zero-SCC is unique, and this is achieved by showing that the zero-state of each zero-SCC satisfying the assertions is reachable from the initial state by a path labeled by v such that $\langle v \rangle_{r,n}$ is in F and does not belong to any proper face of F. Once this is proved, the claim is then a direct consequence of Lemma 182.

Let $q'_{\mathtt{z}}$ be the zero-state of a zero-SCC $\mathcal{S}'$ such that

- there exists an encoding $v \in (\Sigma_r^n)^+ \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q'_{\mathtt{z}})$ with $\langle v \rangle_{r,n} \in F$, and
- $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \mathrm{lin}_{\mathbb{Q}}(\{\langle ou \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q'_{\mathtt{z}} \to q'_{\mathtt{z}})\})$.

Let ou' be the encoding of an element of F which does not belong to any proper face of F. By hypothesis, u' labels a loop rooted at $q'_{\mathtt{z}}$, and therefore we have

$$vu' \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q'_{\mathtt{z}}).$$

Also, let $\mathbf{A}'\mathbf{x} \leq \mathbf{0}$ be the largest subsystem of inequations of $\mathbf{A}\mathbf{x} \leq \mathbf{0}$ such that for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{x} \in F \Rightarrow \mathbf{A}'\mathbf{x} = \mathbf{0}$. By definition, we have

$$F = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0} \wedge \mathbf{A}'\mathbf{x} = \mathbf{0}\},$$

and for each proper face F' of F, there is at least one inequation $\mathbf{a}.\mathbf{x} \leq 0$ from the system $\mathbf{A}\mathbf{x} \leq \mathbf{0}$ but not in $\mathbf{A}'\mathbf{x} \leq \mathbf{0}$ such that for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{x} \in F' \Rightarrow \mathbf{a}.\mathbf{x} = 0.$$

Since $\langle v \rangle_{r,n}$, and $\langle ou' \rangle_{r,n}$ are in F, and since $\langle vu' \rangle_{r,n} = r^{|u'|}\langle v \rangle_{r,n} + \langle ou' \rangle_{r,n}$, we deduce that $\langle vu' \rangle_{r,n}$ is in F.

In addition, by hypothesis, for each inequation $\mathbf{a}.\mathbf{x} \leq 0$ in $\mathbf{A}\mathbf{x} \leq \mathbf{0}$ but not in $\mathbf{A}'\mathbf{x} \leq \mathbf{0}$, we have $\mathbf{a}.\langle v \rangle_{r,n} \leq 0$ and $\mathbf{a}.\langle ou' \rangle_{r,n} < 0$. Therefore, by definition of the encoding scheme, we have

$$\mathbf{a}.\langle vu' \rangle_{r,n} < 0$$

and thus $\langle vu' \rangle_{r,n}$ does not belong to any proper face of F. □

Remark 184. *For any zero-SCC $\mathcal{S}$ and face F of C, if $\mathbf{B}$ is a representative matrix of $\mathcal{S}$ and if $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \mathrm{lin}_{\mathbb{Q}}(\{\langle ou \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\})$ where $q_{\mathtt{z}}$ is the zero-state of $\mathcal{S}$, then, by definition of a representative matrix*

$$\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}.$$

□

Combining the fact that each zero-SCC $\mathcal{S}$ is associated to one and only one face $F_{\mathcal{S}}$ of C as detailed in Theorem 180, and conversely, that each face F of C is associated to one and only one zero-SCC $\mathcal{S}_F$ as detailed in Theorem 183, we deduce the following theorems regarding the linear hulls of C and of the facets of C. The motivation behind those theorems is to be able to recover from the representative matrices of the zero-SCCs of dimension $\dim(C)$ and $\dim(C)-1$ a set of inequations $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$ such that the system $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$ is equivalent to $\mathbf{C}\mathbf{x} \leq \mathbf{0}$. The algorithm is given in the next section.

Theorem 185. *For each zero-SCC $\mathcal{S}$, the dimension of $\mathcal{S}$ is at most* $\dim(C)$.

Proof. Direct consequence of the definition of the definition of a zero-SCC and of Theorem 180. □

Theorem 186. *There is one and only one zero-SCC $\mathcal{S}$ of dimension* $\dim(C)$ *and if* $\mathbf{B}$ *is a representative matrix of $\mathcal{S}$, we have*

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(C).$$

Proof. From Theorem 183, there is one and only one zero-SCC $\mathcal{S}$ such that if $\mathbf{B}$ is a representative matrix of $\mathcal{S}$ and $q_{\mathtt{z}}$ is its zero-state, we have

- $\mathrm{lin}_{\mathbb{Q}}(C) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$, and
- there is an encoding $v \in (\Sigma_r^n)^+$ such that $\langle v \rangle_{r,n} \in C$ and $v \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$.

In addition, thanks to Theorem 180, there is a face F associated to $\mathcal{S}$ such that

$$\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}. \tag{7.29}$$

Since F is a face of C, by definition we have $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \mathrm{lin}_{\mathbb{Q}}(C)$, and we deduce that $\mathrm{lin}_{\mathbb{Q}}(F) = \mathrm{lin}_{\mathbb{Q}}(C)$, i.e.

$$\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}. \tag{7.30}$$

Let $\mathcal{S}'$ be a zero-SCC with a representative matrix $\mathbf{B}'$ such that $\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}'\mathbf{x} = \mathbf{0}\}$. Thanks to Theorem 156, for all encodings u labeling paths from $q_{\mathtt{I}}$ to the zero-state of $\mathcal{S}'$, $\langle u \rangle_{r,n} \in C$, and therefore, thanks to Theorem 183, $\mathcal{S}' = \mathcal{S}$. □

Theorem 187. *Let* $\mathbf{c}.\mathbf{x} \leq 0$ *be an inequation in* $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ *and let F be the facet of C such that $F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\}$. At least one on the following assertions holds.*

- *There exists a zero-SCC $\mathcal{S}$ whose dimension is $\dim(C) - 1$ and such that for any representative matrix $\mathbf{B}$ of $\mathcal{S}$, $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Bx} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(F)$.*
- *For all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{x} \geq \mathbf{0} \Rightarrow \mathbf{c}.\mathbf{x} \leq 0$.*

Proof. Thanks to Theorem 180, there exists one zero-SCC $\mathcal{S}$ such that if $q_{\mathtt{z}}$ is its zero-state and $\mathbf{B}$ is a representative matrix of $\mathcal{S}$, we have

- $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Bx} = \mathbf{0}\}$, and
- there is an encoding $v \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that $\langle v \rangle_{r,n} \in F$.

If $\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Bx} = \mathbf{0}\}$, then the first assertion hold.

Suppose on the other hand that $\mathrm{lin}_{\mathbb{Q}}(F) \subset \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Bx} = \mathbf{0}\}$. Since $\dim(F) = \dim(C) - 1$, by definition of the dimension of a zero-SCC and thanks to Theorem 185, the dimension of $\mathcal{S}$ is $\dim(C)$. Thanks to Theorem 186,

$$\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Bx} = \mathbf{0}\}. \tag{7.31}$$

We prove by contradiction that for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{x} \geq \mathbf{0} \Rightarrow \mathbf{c}.\mathbf{x} \leq 0.$$

Suppose that the assertion does not hold. There would exist $\mathbf{y} \in \mathbb{Q}^n$ with $\mathbf{y} \geq \mathbf{0}$ and $\mathbf{C}^{=}\mathbf{y} = \mathbf{0}$ but $\mathbf{c}.\mathbf{y} > 0$. Since each component of $\mathbf{y}$ is a rational $\frac{n_i}{d_i}$, one could multiply $\mathbf{y}$ by the lowest common multiple of $d_1, \ldots, d_n$ and obtain a positive integer vector, and therefore, there would exist a word $u \in (\Sigma_r^n)^*$ such that $\mathbf{C}^{=}\langle ou \rangle_{r,n} = \mathbf{0}$ and $\mathbf{c}.\langle ou \rangle_{r,n} > 0$. Thanks to Lemma 28, we have

$$\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0}\}.$$

So, given (7.31) and by definition of a representative matrix, u would label a loop rooted at $q_{\mathtt{z}}$. By hypothesis, $\langle v \rangle_{r,n} \in F$, and we have $\mathbf{c}.\langle v \rangle_{r,n} = 0$. Also, v labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$, and thus, for all $k \in \mathbb{N}$, vu^k would label a path from $q_{\mathtt{I}}$ to $q_{\mathtt{z}}$. Let $w \in (\Sigma_r^n)^*$ be a word labeling a path from $q_{\mathtt{z}}$ to an accepting state of $\mathcal{A}$. By definition, $vu^k w$ would belong to $L(\mathcal{A})$ for all $k \in \mathbb{N}$. By definition of the encoding scheme, $\langle vu \rangle_{r,n} = r^{|u|}\langle v \rangle_{r,n} + \langle ou \rangle_{r,n}$, and we would have

$$\mathbf{c}.\langle vu \rangle_{r,n} > \mathbf{c}.\langle v \rangle_{r,n}.$$

Thanks to Lemma 84, we deduce that for all b, there would exist k_b such that

$$\mathbf{c}.\langle vu^{k_b} w \rangle_{r,n} > b.$$

However, this leads to a contradiction. Indeed, from Theorem 21, $P = Q + C$ for some polytope Q, and therefore, $\exists b_{\max} \in \mathbb{Q}$ such that for all $\mathbf{x} \in P$, we have $\mathbf{c}.\mathbf{x} \leq b_{\max}$. □

Theorem 188. *Let $\mathcal{S}$ be a zero-SCC whose dimension is $\dim(C) - 1$ and let $\mathbf{B}$ be a representative matrix of $\mathcal{S}$.*

There exists one and only one facet F such that

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \text{lin}_{\mathbb{Q}}(F).$$

Proof. Thanks to Theorem 180, there exists one and only one face F of C such that

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \text{lin}_{\mathbb{Q}}(F).$$

So, by definition $\dim(F) = \dim(C) - 1$ and, thanks to Theorem 27, F is a facet. □

Theorem 189. *Let $\mathcal{S}$ be a zero-SCC with $\dim(\mathcal{S}) = \dim(C) - 1$, and let $q_{\mathtt{z}}$ be its zero-state and $\mathbf{B}$ be a representative matrix of $\mathcal{S}$.*

There is one and only one inequation $\mathbf{c}.\mathbf{x} \leq 0 \in \mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ such that

1. *Each row of $\mathbf{B}$ is a linear combination of the rows of $\mathbf{C}^=$ and of $\mathbf{c}^{\mathtt{t}}$,*
2. *For all rows $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ either $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \leq 0$ or $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \geq 0$.*
3. *There is a row $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ such that $\mathbf{a}^{\mathtt{t}}$ is not a linear combination of the rows of $\mathbf{C}^=$,*
4. *For all rows $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ such that $\mathbf{a}^{\mathtt{t}}$ is not a linear combination of the rows of $\mathbf{C}^=$, either for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^=\mathbf{x} = \mathbf{0} \ \wedge \ \mathbf{a}.\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^=\mathbf{x} = \mathbf{0} \ \wedge \ \mathbf{c}.\mathbf{x} \leq 0$, or for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^=\mathbf{x} = \mathbf{0} \ \wedge \ (-\mathbf{a}).\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^=\mathbf{x} = \mathbf{0} \ \wedge \ \mathbf{c}.\mathbf{x} \leq 0$.*

Proof. Let $\mathbf{c}_i.\mathbf{x} \leq 0$, $i \in \{1, \ldots, t\}$, be the inequations in $\mathbf{C}^=\mathbf{x} \leq \mathbf{0}$.

Since $\dim(\mathcal{S}) = \dim(C) - 1$, from Theorem 188, there is one and only one facet F such that

$$\text{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}. \tag{7.32}$$

In addition, from Theorem 26 and from Lemma 29, there is an inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ such that

$$F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\} \tag{7.33}$$

$$\text{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^=\mathbf{x} = 0 \wedge \mathbf{c}.\mathbf{x} = 0\} \tag{7.34}$$

1. From (7.32) and (7.34), we deduce that for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{Bx} = \mathbf{0} \Leftrightarrow \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0. \tag{7.35}$$

   Therefore, each row of $\mathbf{B}$ is a linear combination of the rows of $\mathbf{C}^{=}$ and of $\mathbf{c}^{\mathtt{t}}$.

2. Let $\mathbf{a}^{\mathtt{t}}$ be a row of $\mathbf{B}$. As stated above, $\mathbf{a}^{\mathtt{t}}$ is a linear combination of $\{\mathbf{c}_1{}^{\mathtt{t}}, \ldots, \mathbf{c}_t{}^{\mathtt{t}}, \mathbf{c}^{\mathtt{t}}\}$, i.e. $\mathbf{a} = \sum_{i\in\{1,\ldots,t\}} k_i\mathbf{c}_i + k\mathbf{c}$, with $k_1, \ldots, k_t, k \in \mathbb{Q}$. Therefore, for all $\mathbf{x} \in C$, by definition of $\mathbf{C}^{=}$, $\mathbf{C}^{=}\mathbf{x} = \mathbf{0}$, i.e. $\mathbf{c}_i.\mathbf{x} = 0$ for $i \in \{1, \ldots, t\}$, and $\mathbf{a}.\mathbf{x} = k \cdot \mathbf{c}.\mathbf{x}$. Since $\mathbf{x} \in C$, $\mathbf{c}.\mathbf{x} \leq 0$, and therefore, we conclude that either $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \leq 0$ or $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \geq 0$.

3. There is a row $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ such that $\mathbf{a}^{\mathtt{t}}$ is not a linear combination of the rows of $\mathbf{C}^{=}$. Otherwise, $\mathbf{c}^{\mathtt{t}}$ would be linear combination of the rows of $\mathbf{C}^{=}$ since $\mathbf{Bx} = 0 \Rightarrow \mathbf{c}.\mathbf{x} = 0$, and $\mathbf{c}.\mathbf{x} = 0$ would be an implicit equation in C, i.e. $\mathbf{c}.\mathbf{x} \leq 0$ would be in $\mathbf{C}^{=}\mathbf{x} \leq \mathbf{0}$, violating the fact that $\mathbf{c}.\mathbf{x} \leq 0$ is in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$.

4. For all rows $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ such that $\mathbf{a}^{\mathtt{t}}$ is not a linear combination of the rows of $\mathbf{C}^{=}$, $\mathbf{a} = \sum_{i\in\{1,\ldots,t\}} k_i\mathbf{c}_i + k\mathbf{c}$, with $k_1, \ldots, k_m, k \in \mathbb{Q}$ and $k \neq 0$. If $k > 0$, for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^{=}\mathbf{x} = 0 \wedge \mathbf{a}.\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^{=}\mathbf{x} = 0 \wedge \mathbf{c}.\mathbf{x} \leq 0$, and if $k < 0$, for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge (-\mathbf{a}).\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} \leq 0$. □

7.2.4 Algorithm

In this section, we present an algorithm that, given reduced minimal NDD $\mathcal{A}$ representing the integer elements of a polyhedron $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Ax} \leq \mathbf{b}\}$ such that $P \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$, synthesizes a system of linear inequations corresponding to the characteristic cone C of P. The algorithm uses the properties given in Section 7.2.3 regarding the zero-SCCs of dimension $d_{\max}$ and $d_{\max} - 1$ where $d_{\max}$ is the dimension of the characteristic cone, and its overall time complexity is polynomial with respect to the size of the input NDD.

Recall that if the characteristic cone is described via the system of linear inequations $\mathbf{Cx} \leq \mathbf{0}$ containing no redundant inequation, the system can be partitioned into a system of implicit equations $\mathbf{C}^{=}\mathbf{x} \leq \mathbf{0}$ and the other inequations $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$, and there is a bijection between the facets of C and the inequations in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$ such that F is a facet of C if and only if $F = \{\mathbf{x} \in C \mid \mathbf{a}.\mathbf{x} \leq 0\}$ for some inequation $\mathbf{a}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$. The idea of the algorithm is to compute

a system of linear inequations equivalent to $\mathbf{C}^{=}\mathbf{x} = \mathbf{0}$ and for each facet F, an inequation $\mathbf{a}'.\mathbf{x} \leq 0$ such that $F = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{a}'.\mathbf{x} \leq 0\}$.

The algorithm works as follows. It first extracts the zero-SCCs. Thanks to Theorem 185, the dimensions of the zero-SCCs are at most $\dim(C)$, and thanks to Theorem 186, there is exactly one zero-SCC, $\mathcal{S}_{\max}$, whose dimension is $\dim(C)$. Also, if $\mathbf{B}_{\max}$ is the computed representative matrix for $\mathcal{S}_{\max}$, we have

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}_{\max}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(C). \tag{7.36}$$

One initializes the matrix $\mathbf{C}'$ to $\begin{bmatrix} -\mathbf{I}_n \\ \mathbf{B}_{\max} \\ -\mathbf{B}_{\max} \end{bmatrix}$. Recall that $\mathbf{I}_n$ is the $n \times n$ identity matrix. So, at this point, for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\mathbf{C}'\mathbf{x} \leq \mathbf{0} \text{ iff } \mathbf{x} \in \mathrm{lin}_{\mathbb{Q}}(C) \wedge \mathbf{x} \geq \mathbf{0}.$$

Recall that for all facets F of C, $F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\}$ for some inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$. Thanks to Theorem 187, for each inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$, either

- for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{x} \geq \mathbf{0} \Rightarrow \mathbf{c}.\mathbf{x} \leq 0$, or
- there exists a zero-SCC $\mathcal{S}$ whose dimension is $\dim(C) - 1$ and such that for any representative matrix $\mathbf{B}$ of $\mathcal{S}$, $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(F)$.

Clearly, if the first assertion holds, then at this stage of the algorithm, we have $\mathbf{C}'\mathbf{x} \leq \mathbf{0} \Rightarrow \mathbf{c}.\mathbf{x} \leq 0$ for all $\mathbf{x} \in \mathbb{Q}^n$.

In order to cover all inequations in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$, one has to consider all zero-SCCs of dimension $\dim(C) - 1$, and this is done via the recursive procedure EXPLORE-BW which is basically a backward search, starting at the zero-state of $\mathcal{S}_{\max}$ such that each state is visited at most once. Its arguments are a state q and a word w such that w labels a path from q to the zero-state of $\mathcal{S}_{\max}$. It first marks the state q as visited. If q is a zero-state, one checks whether the dimension of the corresponding zero-SCC $\mathcal{S}$ is $d_{\max} - 1$. If this is the case, then thanks to Theorem 189, there is one and only one inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$ such that if $\mathbf{B}$ is a representative matrix of the zero-SCC having q as zero-state then for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{B}\mathbf{x} = \mathbf{0} \Leftrightarrow \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} = 0$. Also, it is possible to extract a row $\mathbf{a}^{\mathsf{t}}$ from $\mathbf{B}$ such that either for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{a}.\mathbf{x} \leq 0 \text{ iff } \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} \leq 0,$$

or for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge -\mathbf{a}.\mathbf{x} \leq 0 \text{ iff } \mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} \leq 0.$$

One checks which assertion holds by testing the sign of $\mathbf{a}.\langle ow \rangle_{r,n}$. Indeed, thanks to Lemma 159 and by definition of $\mathbf{a}$, for all encodings v labeling a path from $q_{\mathtt{I}}$ to q, we have $\mathbf{a}.\langle v \rangle_{r,n} = 0$, $\mathbf{C}^{=}\langle vw \rangle_{r,n} = \mathbf{0}$ and $\mathbf{a}.\langle vw \rangle_{r,n} \neq 0$ (since $\mathbf{a}^{\mathtt{t}}$ is not a linear combination of the rows of $\mathbf{C}^{=}$). Therefore, by definition of the encoding scheme, we have

$$\mathbf{a}.\langle ow \rangle_{r,n} \neq 0.$$

Once $\mathbf{a}$ is identified, one adds $\mathbf{a}^{\mathtt{t}}$ (or $-\mathbf{a}^{\mathtt{t}}$ depending on which assertion holds) as a new row of the matrix $\mathbf{C}'$.

The procedure terminates with a recursive call, propagating the current path backward. When all states have been explored, the function returns the system of linear inequations $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$.

By construction, for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\mathbf{C}\mathbf{x} \leq \mathbf{0} \Rightarrow \mathbf{C}'\mathbf{x} \leq \mathbf{0}.$$

Also, for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\mathbf{C}'\mathbf{x} \leq \mathbf{0} \Rightarrow \mathbf{C}^{=}\mathbf{x} \leq \mathbf{0},$$

and thanks to Theorems 187 and 188, for all inequations $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^{+}\mathbf{x} \leq 0$, for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{C}'\mathbf{x} \leq \mathbf{0} \Rightarrow \mathbf{c}\mathbf{x} \leq 0.$$

So, we conclude that for all $\mathbf{x} \in \mathbb{Q}^n$,

$$\mathbf{C}'\mathbf{x} \leq \mathbf{0} \Leftrightarrow \mathbf{C}\mathbf{x} \leq \mathbf{0}.$$

A formal description of the above function, called CHARCONEFORMULA, is given in Fig.7.4. Based on the above description, we have the following theorem.

Theorem 190. *Let $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be the reduced minimal NDD representing the positive integer elements of a polyhedron $P \subseteq \mathbb{Q}^n$.*

Let $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$ be the system of linear inequations returned by the function CHARCONEFORMULA*, described in Fig.7.4.*

Then $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$ is such that

$$\text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}'\mathbf{x} \leq \mathbf{0}\}.$$

The time complexity of CHARCONEFORMULA *is $\mathcal{O}(|\delta| \cdot n^2 + k \cdot |Q| \cdot n)$ where k is the number of zero-SCCs whose dimension is $\dim(\text{char-cone}(P)) - 1$.*

Proof. The correctness has already been partly justified in the informal description, and we detail below the complexity of the algorithm.

The computation of the zero-SCC can be achieved in time linear with respect to $|\mathcal{A}|$. Indeed, one simply has to check the states having a loop labeled by o and then compute the maximal strongly connected components to which those states belong. The computation of these SCCs can be done in linear time with respect of $|\mathcal{A}|$ with standard algorithms [Tar72].

Thanks to Theorem 177, the dimension of a zero-SCC is provided by the number of incoming transitions in any state of the same SCC rooted at states in the same SCC, and therefore, it can be computed in constant time.

Thanks to Theorem 175, there exists an algorithm GETREPRESENTATIVEMATRIX which, given the reduced minimal NDD $\mathcal{A}$ and the zero-state of a zero-SCC $\mathcal{S}(Q_{\mathcal{S}}, \delta_{\mathcal{S}})$ of $\mathcal{A}$, computes a representative matrix of $\mathcal{S}$ and whose time complexity is $\mathcal{O}(|\delta_{\mathcal{S}}| \cdot n^2)$ where $|\delta_{\mathcal{S}}|$ denotes the number of transitions in $\mathcal{S}$.

Since the states visited during the backward search are marked as visited when first met and since one extends the search only for states not yet visited, each state q appears at most once as argument of EXPLORE-BW. Also, one easily proves by recursion that if the state q and the word w are arguments of EXPLORE-BW, then w labels a path from q to the zero-state of the zero-SCC $\mathcal{S}_{\max}$.

Finally, thanks to Theorem 163, $\mathcal{S}_{\max}$ is reachable from all zero-SCCs $\mathcal{S}$, and in particular from all zero-SCCs of dimension $d_{\max} - 1$. So, the zero-states of all zero-SCCs of dimension $d_{\max} - 1$ are handled in the recursive called EXPLORE-BW. If they are m zero-SCC of dimension $d_{\max} - 1$, the overall cost of the backward search is $\mathcal{O}(|\delta| + m \cdot |Q| \cdot n)$.

Adding the costs involved at each step of the computation, the overall time complexity of the function CHARCONEFORMULA is $\mathcal{O}(|\delta| \cdot n^2 + m \cdot |Q| \cdot n)$. □

Example 191. *In the NDD $\mathcal{A}_{\mathtt{x}}$ displayed in Fig.7.2, the zero-SCC with the maximal dimension is the one associated to the zero-state q_3. Its dimension is 2 and the associated vector space is $\mathbb{Q}^2$. A representative matrix is $\begin{bmatrix} 0 & 0 \end{bmatrix}$. There are two zero-SCCs with dimension 1, those associated to the zero-states q_2 and q_5. The associated vector spaces are $\{(x, y) \in \mathbb{Q}^2 \mid x - y = 0\}$ and $\{(x, y) \in \mathbb{Q}^2 \mid x - 2 \cdot y = 0\}$. A representative matrice for the zero-SCC associated to q_2 is $\begin{bmatrix} 1 & -1 \end{bmatrix}$ and a representative matrix for the zero-SCC associated to q_5 is $\begin{bmatrix} 1 & -2 \end{bmatrix}$. Given the labels of the paths from q_2 and q_5 to q_3, the characteristic cone is $\{(x, y) \in \mathbb{Q}^2 \mid x \geq 0 \wedge y \geq 0 \wedge -x + y \leq 0 \wedge x - 2 \cdot y \leq 0\}$.* □

function CHARCONEFORMULA(NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$) : system of linear inequations

```
 1:  var S_SCC : set of zero-SCC;
 2:      Q_visited : set of state;
 3:      d_max : integer;
 4:      B,C' : array of rational;
 5:  procedure EXPLORE-BW(state q, word w)
 6:      var q' : state;
 7:          α : symbol;
 8:          S :  zero-SCC;
 9:          a : vector of rational;
10:      begin
11:          Q_visited := Q_visited ∪ {q};
12:          if q is a zero-state then
13:              begin
14:                  let S ∈ S_SCC such that q is zero-state of S;
15:                  if GETDIMENSIONSCC(S) = d_max − 1 then
16:                      begin
17:                          B := GETREPRESENTATIVEMATRIX(S);
18:                          let a such that a^t is a row of B
                                                  and a.⟨ow⟩_{r,n} ≠ 0;
19:                          if a.⟨ow⟩_{r,n} > 0 then a := −a;
20:                          C' := [ C' ]
                                   [ a^t ];
21:                      end
22:              end
23:          for each α ∈ Σ_r^n, q' ∈ Q \ Q_visited such that δ(q', α) = q do
24:              EXPLORE-BW(q', αw);
25:      end

    (...)
```

Figure 7.4: Function CHARCONEFORMULA

(...)

begin

S_{SCC} := GETZEROSCCS($\mathcal{A}$);

$d_{\max}$:= $\max_{\mathcal{S}\in S_{SCC}}$(GETDIMENSIONSCC ($\mathcal{S}$);

let $\mathcal{S}_{\max}$ **such that** GETDIMENSIONSCC($\mathcal{S}_{\max}$) = $d_{\max}$;

$\mathbf{B}$:= GETREPRESENTATIVEMATRIX($\mathcal{S}_{\max}$);

$$\mathbf{C}' := \begin{bmatrix} -\mathbf{I}_n \\ \mathbf{B} \\ -\mathbf{B} \end{bmatrix};$$

EXPLORE-BW(GETZEROSTATE($\mathcal{S}_{\max}$), ε);

return $\mathbf{C}'\mathbf{x} \leq \mathbf{0}$;

end

Figure 7.5: Function CHARCONEFORMULA (continued)

7.3 Synthesis of Formula for char-cone(P)

In this section, we generalize the results presented in Section 7.2 as follows. In Sections 7.2, all integer elements of the polyhedra had to be in $\mathbb{N}^n$ i.e. $\text{sign}_r(\mathbf{z}) = o$ for all $\mathbf{z} \in P \cap \mathbb{Z}^n$. In this section, we still impose that all integer elements have the same sign, but this sign is arbitrary, i.e. $\text{sign}_r(\mathbf{z}_1) = \alpha_{\texttt{sign}} = \text{sign}_r(\mathbf{z_2})$ for all $\mathbf{z}_1, \mathbf{z}_2 \in P \cap \mathbb{Z}^n$ and some sign symbol $\alpha_{\texttt{sign}}$.

Basically, all the results of Section 7.2 are modified in the following way. All encodings labeling paths in the NDD will have the same sign symbol $\alpha_{\texttt{sign}}$. The concept of zero-loop is generalized to the concept of sign-loop, that is, simple loop labeled by $\alpha_{\texttt{sign}}$. Also, the role played by the faces of char-cone(P) are now played by the faces of the polyhedron $\langle\alpha_{\texttt{sign}}\rangle_{r,n} + \text{char-cone}(P)$.

Throughout this section, $\mathcal{A} = (\Sigma_r^n, Q, \delta, q_{\texttt{I}}, Q_{\texttt{F}})$ denotes the reduced minimal NDD accepting the encoding in base r of the elements in the set $P \cap \mathbb{Z}^n$, where $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ for some integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ and integer vector $\mathbf{b} \in \mathbb{Z}^m$. We impose the additional condition that $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}\}$.

The inequations in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ are $\mathbf{a}_1.\mathbf{x} \leq b_1, \ldots, \mathbf{a}_m.\mathbf{x} \leq b_m$. Also, the characteristic cone of P is denoted by C, i.e.

$$C = \text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq 0\}.$$

Finally, $\mathbf{C}$ is an integer matrix such that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Cx} \leq \mathbf{0}\}$ and such that no inequation in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ is redundant in $\mathbf{Cx} \leq \mathbf{0}$. The system of inequations $\mathbf{Cx} \leq \mathbf{0}$ can be generated by removing successively the redundant inequations in $\mathbf{Ax} \leq \mathbf{0}$.

We have the following lemma.

Lemma 192. *For all non-empty words u labeling a path rooted at $q_{\mathtt{I}}$, we have*

- $\mathrm{sign}_r(\langle u \rangle_{r,n}) = \alpha_{\mathtt{sign}}$,
- $\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} = \langle u \rangle_{r,n}$, *and,*
- $\hat{\delta}(q_{\mathtt{I}}, u) = \hat{\delta}(q_{\mathtt{I}}, \alpha_{\mathtt{sign}} u)$.

Proof. Similar to the proof of Lemma 151. □

We highlight an important property shared by the elements in C. The proof also provides some valuable insights on how one handles decomposition of encodings having $\alpha_{\mathtt{sign}}$ as sign symbol.

Lemma 193. *For all integer elements $\mathbf{x}$ in $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$, we have $\mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}$.*

Proof. By definition, $\mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}$ if and only if, for all $i \in \{1, \ldots n\}$,

$$\begin{aligned} \mathbf{x}[i] < 0 \quad &\text{if} \quad \alpha_{\mathtt{sign}}[i] = r - 1, \textit{ and} \\ \mathbf{x}[i] \geq 0 \quad &\text{if} \quad \alpha_{\mathtt{sign}}[i] = 0. \end{aligned}$$

In addition, by hypothesis, $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Ax} \leq \mathbf{b}\}$ and $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$.

Let $i \in \{1, \ldots n\}$.

- Suppose $\alpha_{\mathtt{sign}}[i] = r - 1$. Then by hypothesis, we have

  $$P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Ax} \leq \mathbf{b} \wedge \mathbf{x}[i] < 0\}.$$

  Therefore, by definition, we have

  $$C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{Ax} \leq \mathbf{0} \wedge \mathbf{x}[i] \leq 0\},$$

  and thus

  $$\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) \leq \mathbf{0} \wedge \mathbf{x}[i] \leq -1\}.$$

  We deduce that for all $\mathbf{x} \in \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (C \cap \mathbb{Z}^n)$, we have $\mathbf{x}[i] < 0$.

- Suppose $\alpha_{\texttt{sign}}[i] = 0$. Then by hypothesis, we have

$$P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b} \wedge \mathbf{x}[i] \geq 0\}.$$

  Therefore, by definition, we have

$$C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0} \wedge \mathbf{x}[i] \geq 0\},$$

  and thus

$$\langle \alpha_{\texttt{sign}} \rangle_{r,n} + C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}(\mathbf{x} - \langle \alpha_{\texttt{sign}} \rangle_{r,n}) \leq \mathbf{0} \wedge \mathbf{x}[i] \geq 0\}.$$

  We deduce that for all $\mathbf{x} \in (C \cap \mathbb{Z}^n) + \langle \alpha_{\texttt{sign}} \rangle_{r,n}$, we have $\mathbf{x}[i] \geq 0$.

So, we conclude that for all $\mathbf{x} \in (C \cap \mathbb{Z}^n) + \langle \alpha_{\texttt{sign}} \rangle_{r,n}$, $\text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}$. □

From the above lemma, we deduce that the characteristic cone C is pointed.

Lemma 194. *The cone C is pointed.*

Proof. Let $\mathbf{x} \in C \cap (-C)$. By definition, $\mathbf{x} \in \mathbb{Q}^n$ and there exists a vector $\mathbf{y} \in \mathbb{Z}^n$ such that $\mathbf{y} = k \cdot \mathbf{x}$ for some positive integer k. Since C and $(-C)$ are cones, $\mathbf{y} \in C$.

Since $\mathbf{y} \in C \cap \mathbb{Z}^n$, thanks to Lemma 193, for all $i \in \{1, \ldots, n\}$, we have

$$\mathbf{y}[i] \leq 0 \quad \text{if} \quad \langle \alpha_{\texttt{sign}} \rangle_{r,n}[i] = r - 1 \tag{7.37}$$
$$\mathbf{y}[i] \geq 0 \quad \text{if} \quad \langle \alpha_{\texttt{sign}} \rangle_{r,n}[i] = 0 \tag{7.38}$$

Similarly, since $\mathbf{y} \in (-C)$, we have $-\mathbf{y} \in C$ and

$$-\mathbf{y}[i] \leq 0 \quad \text{if} \quad \langle \alpha_{\texttt{sign}} \rangle_{r,n}[i] = r - 1 \tag{7.39}$$
$$-\mathbf{y}[i] \geq 0 \quad \text{if} \quad \langle \alpha_{\texttt{sign}} \rangle_{r,n}[i] = 0 \tag{7.40}$$

We deduce that for all $i \in \{1, \ldots, n\}$, $\mathbf{y}[i] = 0$. We conclude that $C \cap (-C) = \{\mathbf{0}\}$, i.e. C is pointed. □

Example 195. *We give in Fig.7.6 the reduced minimal NDD $\mathcal{A}_{\texttt{sign}}$ representing the set $S_{\texttt{sign}}$, with*

$$S_{\texttt{sign}} = \{(x, y) \in \mathbb{Z}^2 \mid x - y \leq 1 \wedge x - 2y \geq 2 \wedge x < 0 \wedge y < 0\}. \tag{7.41}$$

We will use this example to illustrate some definitions and theorems throughout this chapter.

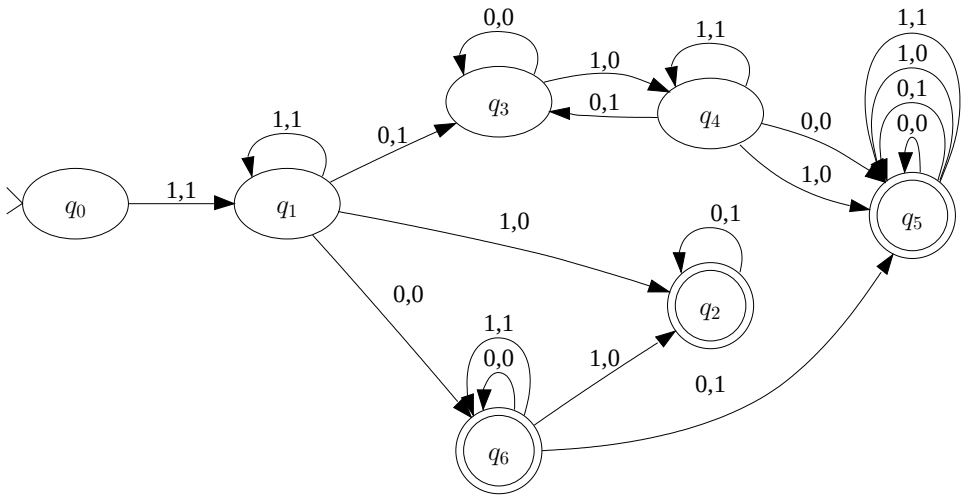


Figure 7.6: minimal reduced NDD $\mathcal{A}_{\mathtt{sign}}$ representing $S_{\mathtt{sign}}$

Since the sign symbol of elements in $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C \cap \mathbb{Z}^n$ and in $P \cap \mathbb{Z}^n$ is $\alpha_{\mathtt{sign}}$, and given the role played by the elements in those sets, the decomposition $\langle ouv \rangle_{r,n} = r^{|v|}\langle ou \rangle_{r,n} + \langle ov \rangle_{r,n}$, which appeared in many technical proofs of Section 7.2, will be substituted in this section by the decomposition $\langle \alpha_{\mathtt{sign}} uv \rangle_{r,n} = r^{|v|}\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} - r^{|v|}\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \langle \alpha_{\mathtt{sign}} v \rangle_{r,n}$. In order to assess the usefulness of this decomposition, if $\mathbf{a}.\mathbf{x} \leq b$ is an inequation satisfied by all elements in the polyhedron P, then for all elements in C, $\mathbf{a}.\mathbf{x} \leq 0$, and therefore, for all elements $\langle \alpha_{\mathtt{sign}} u \rangle_{r,n}$ in $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C \cap \mathbb{Z}^n$, $\mathbf{a}.(\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) \leq 0$. So, based on the decomposition presented above, we have $\mathbf{a}.\langle \alpha_{\mathtt{sign}} uv \rangle_{r,n} = \mathbf{a}.\langle \alpha_{\mathtt{sign}} v \rangle_{r,n}$. This simple example already introduces the fact that, in some important cases, we will be able to deal only with the suffices of the encodings labeling paths rooted at $q_{\mathtt{I}}$, dropping the prefix (except the sign symbol), exactly as we did in Section 7.2.

7.3.1 Sign-states

Any simple loop in $\mathcal{A}$ labeled by a sequence of $\alpha_{\mathtt{sign}}$ must be of size 1. We call such a loop *a sign-loop*, and the state at which the loop is rooted a *sign-state*. The concepts of sign-loop and sign-state generalize the concept of zero-loop and zero-state introduced in Section 7.2.1.

We show that the properties of zero-states translate easily to similar properties of sign-states. Generally, it suffices to replace the symbol o by $\alpha_{\mathtt{sign}}$ and the cone C by the polyhedron $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$. The reason of this is rooted in the encoding

scheme and in the fact that $\alpha_{\texttt{sign}}$ is the sign symbol of all encodings labeling paths rooted at $q_{\texttt{I}}$.

Lemma 196. *Any simple loop in $\mathcal{A}$ labeled by a sequence of $\alpha_{\texttt{sign}}$ is of size* 1.

Proof. Similar to proof of Lemma 153. □

Example 197. *In the NDD $\mathcal{A}_{\texttt{sign}}$ of Fig.7.6, the sign symbol $\alpha_{\texttt{sign}}$ is $(1, 1)$ and there are four sign-states, q_1, q_4, q_5 and q_6.*

We now show a relationship between the integer elements in $\langle\alpha_{\texttt{sign}}\rangle_{r,n} + C$ and those in P, generalizing Lemma 155.

Lemma 198. *For all encodings $u \in (\Sigma_r^n)^*$, $\langle u\rangle_{r,n}$ is in the polyhedron $\langle\alpha_{\texttt{sign}}\rangle_{r,n}+ C$ if and only if for all elements $\langle\alpha_{\texttt{sign}}v\rangle_{r,n}$ of P, $\langle uv\rangle_{r,n}$ is also in P.*

Proof. Suppose that $\langle u\rangle_{r,n} \in \langle\alpha_{\texttt{sign}}\rangle_{r,n} + C$. Let $v \in (\Sigma_r^n)^*$ with $\langle\alpha_{\texttt{sign}}v\rangle_{r,n} \in P$. We have $\mathbf{A}(\langle u\rangle_{r,n} - \langle\alpha_{\texttt{sign}}\rangle_{r,n}) \leq 0$ and $\mathbf{A}\langle\alpha_{\texttt{sign}}v\rangle_{r,n} \leq \mathbf{b}$. By definition of the encoding scheme, we have

$$\begin{aligned}
\mathbf{A}\langle uv\rangle_{r,n} &= \mathbf{A}(r^{|v|}\langle u\rangle_{r,n} + \langle ov\rangle_{r,n}) \\
&\leq \mathbf{A}(r^{|v|}\langle\alpha_{\texttt{sign}}\rangle_{r,n} + \langle ov\rangle_{r,n}) \\
&\leq \mathbf{A}\langle\alpha_{\texttt{sign}}v\rangle_{r,n} \\
&\leq \mathbf{b}
\end{aligned}$$

Therefore, by definition, $\langle uv\rangle_{r,n} \in P$.

Suppose that $\langle u\rangle_{r,n} \notin \langle\alpha_{\texttt{sign}}\rangle_{r,n}+C$. We have $\mathbf{A}(\langle u\rangle_{r,n}-\langle\alpha_{\texttt{sign}}\rangle_{r,n}) \not\leq \mathbf{0}$. So, there exists an inequation $\mathbf{a}.\mathbf{x} \leq b$ in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ such that $\mathbf{a}.(\langle u\rangle_{r,n}-\langle\alpha_{\texttt{sign}}\rangle_{r,n}) > 0$. Therefore, by definition of the encoding scheme,

$$\begin{aligned}
\mathbf{a}.\langle u\alpha_{\texttt{sign}}\rangle_{r,n} &= r\mathbf{a}.\langle u\rangle_{r,n} + \mathbf{a}.\langle o\alpha_{\texttt{sign}}\rangle_{r,n} \\
&= r\mathbf{a}.\langle u\rangle_{r,n} + \mathbf{a}.((1-r)\langle\alpha_{\texttt{sign}}\rangle_{r,n}) \\
&= \mathbf{a}.\langle u\rangle_{r,n} + (r-1)\mathbf{a}.(\langle u\rangle_{r,n} - \langle\alpha_{\texttt{sign}}\rangle_{r,n}) \\
&> \mathbf{a}.\langle u\rangle_{r,n}
\end{aligned}$$

Thanks to Lemma 85, there exists $k_{\min}$ such that for all words $w \in (\Sigma_r^n)^*$, $\mathbf{a}.\langle u\alpha_{\texttt{sign}}^k w\rangle_{r,n} > b$ for all $k \geq k_{\min}$, and so, there exists a word $v \in (\Sigma_r^n)^*$ such that $\langle\alpha_{\texttt{sign}}v\rangle_{r,n} \in P$ and $\langle uv\rangle_{r,n} \notin P$. □

We now describe the link between paths in $\mathcal{A}$ and the characteristic cone C, and this generalizes Theorem 156.

Theorem 199. *Let Q_C be the set of states q such that there is a path from q to a sign-state labeled by a sequence of $\alpha_{\mathtt{sign}}$ symbols.*

The integer elements of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$ are exactly the integer vectors whose encodings label paths from $q_{\mathtt{I}}$ to a state in Q_C.

Proof. The proof is exactly the same as the proof of Theorem 156. It suffices to substitute $\alpha_{\mathtt{sign}}$ for o, and use Lemmas 198 and 196 instead of Lemmas 155 and 153 respectively. □

Based on the previous theorem, given $\mathcal{A}$, we can generate in time proportional to $|\mathcal{A}|$ a deterministic NDD $\mathcal{A}_C$ accepting the encodings of the integer elements of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$ by setting all states in Q_C as the only accepting states, i.e. $\mathcal{A}_C = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_C)$. This can be done by performing a backward search, starting from all sign-states, and following only transitions labeled by $\alpha_{\mathtt{sign}}$. The states reached are exactly those in Q_C.

We now address the characterization of the languages $L_{\mathcal{A}}(q_{\mathtt{s}} \rightarrow q_{\mathtt{s}})$ and $L_{\mathcal{A}}(q_{\mathtt{s}})$ for any sign-state $q_{\mathtt{s}}$. Again, this is a simple generalization of the corresponding results established for zero-states. We first adapt the definition of a pending inequation.

Definition 200. *An inequation $\mathbf{a}.\mathbf{x} \leq b$ is* pending in *a sign-state $q_{\mathtt{s}}$ if for all words $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{s}})$, $\mathbf{a}.(\langle u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = 0$.*

As for the zero-state, we first prove a lemma regarding words labeling loops rooted at sign-state.

Lemma 201. *For any sign-state $q_{\mathtt{s}}$ in $\mathcal{A}$, if $q_{\mathtt{s}}$ is reachable from $q_{\mathtt{I}}$ by a path labeled by $\alpha_{\mathtt{sign}} u$, then there is a loop rooted at $q_{\mathtt{s}}$ labeled by u.*

Proof. The proof is similar to the proof of Lemma 159. □

Lemma 202. *Let $q_{\mathtt{s}}$ be a sign-state. There is a word $u_{\mathtt{s}} \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{s}})$ such that for all inequations $\mathbf{a}.\mathbf{x} \leq b$ from $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{s}}$, $\mathbf{a}.(\langle u_{\mathtt{s}} \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = 0$ and for all inequations $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ not pending in $q_{\mathtt{s}}$, $\mathbf{a}.\langle u_{\mathtt{s}} \rangle_{r,n} < \min(b, -\| \mathbf{a}^+ \|)$.*

Proof. The proof is similar to the proof of Lemma 160. However, given the small particularities, we give a full proof below.

Recall that $\mathbf{a}_i.\mathbf{x} \leq b_i$, $i \in \{1, \ldots, m\}$ are the inequations of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$. We partition the inequations $\mathbf{a}_i.\mathbf{x} \leq b_i$, $i \in \{1, \ldots, m\}$ into those pending in $q_{\mathtt{s}}$ and

those not pending in $q_{\mathtt{s}}$. Let $I_{\mathtt{p}} \subseteq \{1,\ldots,m\}$ such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{s}}$.

Since the sign symbol of all encodings labeling paths rooted at $q_{\mathtt{I}}$ is $\alpha_{\mathtt{sign}}$ and by definition of $I_{\mathtt{p}}$, for $i \in \{1,\ldots,m\} \setminus I_{\mathtt{p}}$ there is a word $\alpha_{\mathtt{sign}}u_i$ labeling a path from $q_{\mathtt{I}}$ to $q_{\mathtt{s}}$ such that

$$\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) \neq 0.$$

From Theorem 199, $\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n}$ belongs to $\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + C$ and thus we have

$$A(\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) \leq \mathbf{0}.$$

In particular, we have $\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) < 0$. Also, since $\alpha_{\mathtt{sign}}u_i$ labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{s}}$, thanks to Lemma 201, we deduce that

$$u_i \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}}).$$

Let u be $u_1u_2\ldots u_m$ such that for $i \in \{1,\ldots,m\}$, $\alpha_{\mathtt{sign}}u_i$ labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{s}}$, and $\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) < 0$ if $i \notin I_{\mathtt{p}}$. By construction, $\alpha_{\mathtt{sign}}u$ labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{s}}$, and therefore, by definition, for all $i \in I_{\mathtt{p}}$, we have

$$\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) = 0.$$

Also, for all $i \in \{1,\ldots,m\} \setminus I_{\mathtt{p}}$, we have

$$\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) < 0.$$

Indeed, by definition, $\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) < 0$, and for all $j \in \{1,\ldots,m\}$, $\mathbf{a}_i.(\langle \alpha_{\mathtt{sign}}u_j\rangle_{r,n} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) \leq 0$. So, according to Lemma 81, we have

$$\begin{array}{rcl}
\mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u\rangle_{r,n} & = & \mathbf{a}_i.(r^{|u_2\ldots u_m|}\langle \alpha_{\mathtt{sign}}u_1\rangle_{r,n} + \langle ou_2\ldots u_m\rangle_{r,n}) \\
& \leq & \mathbf{a}_i.(r^{|u_2\ldots u_m|}\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + \langle ou_2\ldots u_m\rangle_{r,n}) \\
& \leq & \mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u_2\ldots u_m\rangle_{r,n}
\end{array}$$

Proceeding similarly to remove u_2, …, u_{i-1}, we find

$$\mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u\rangle_{r,n} \leq \mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u_i\ldots u_m\rangle_{r,n}.$$

Then, since $\mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} < \mathbf{a}_i.\langle \alpha_{\mathtt{sign}}\rangle_{r,n}$, we have

$$\begin{array}{rcl}
\mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u_i\ldots u_m\rangle_{r,n} & = & \mathbf{a}_i.(r^{|u_{i+1}\ldots u_m|}\langle \alpha_{\mathtt{sign}}u_i\rangle_{r,n} + \langle ou_{i+1}\ldots u_m\rangle_{r,n} \\
& < & \mathbf{a}_i.(r^{|u_{i+1}\ldots u_m|}\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + \langle ou_{i+1}\ldots u_m\rangle_{r,n} \\
& < & \mathbf{a}_i.\langle \alpha_{\mathtt{sign}}u_{i+1}\ldots u_m\rangle_{r,n}.
\end{array}$$

Again, we can apply similar development to eliminate successively u_{i+1}, ..., u_m, and we find $\mathbf{a}_i.\langle\alpha_{\mathtt{sign}}u_{i+1}\dots u_m\rangle_{r,n} \leq \mathbf{a}_i.\langle\alpha_{\mathtt{sign}}\rangle_{r,n}$.

So, combining the above results, we have $\mathbf{a}_i.\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} < \mathbf{a}_i.\langle\alpha_{\mathtt{sign}}\rangle_{r,n}$.

Finally, from Lemma 85, there exists $k \in \mathbb{N}$ such that for all $i \in \{1,\dots,m\} \setminus I_{\mathtt{p}}$, $\mathbf{a}_i.\langle\alpha_{\mathtt{sign}}u^k\rangle_{r,n} < \min(b_i, -\|\mathbf{a}_i{}^+\|)$ and for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.(\langle\alpha_{\mathtt{sign}}u^k\rangle_{r,n} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = 0$. Since $\alpha_{\mathtt{sign}}u^k \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $u_{\mathtt{s}} = u^k$ satisfies the claim. □

Exactly as we did for pending inequations in zero-state, we deduce that the inequations not pending in a sign-state $q_{\mathtt{s}}$ do not constrain the language accepted from $q_{\mathtt{z}}$. In addition, by definition, for all inequations $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ pending in $q_{\mathtt{s}}$, $\mathbf{a}.(\langle u_{\mathtt{s}}\rangle_{r,n} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = 0$, and therefore, according to the encoding scheme, all suffixes w labeling a path from $q_{\mathtt{s}}$ to an accepting state have to satisfy $\mathbf{a}.\langle\alpha_{\mathtt{sign}}w\rangle_{r,n} \leq b$. Thanks to this result, we can specify $L_{\mathcal{A}}(q_{\mathtt{s}})$.

Theorem 203. *Let $q_{\mathtt{s}}$ be a sign-state and let $I_{\mathtt{p}} \subseteq \{1,\dots,m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{s}}$.*

$$L_{\mathcal{A}}(q_{\mathtt{s}}) = \{w \mid \bigwedge_{i\in I_{\mathtt{p}}} \mathbf{a}_i\langle\alpha_{\mathtt{sign}}w\rangle_{r,n} \leq b_i\}.$$

Proof. The proof is similar to the proof of Theorem 161. □

The next theorem is also a consequence of the fact that only pending inequations play a role in the language accepted from a sign-state. It shows that the language of words labeling loops rooted at some sign-state correspond to the set of encodings (with no sign symbol) of the integer solutions of a system of linear equations, i.e. the integer elements of a $\mathbb{Q}$-affine space.

Theorem 204. *Let $q_{\mathtt{s}}$ be a sign-state and let $I_{\mathtt{p}} \subseteq \{1,\dots,m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{s}}$.*

$$L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}}) = \{u \mid \bigwedge_{i\in I_{\mathtt{p}}} \mathbf{a}_i.(\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = 0\}.$$

Proof. The proof is similar to the proof of Theorem 162. □

Finally, the relationships between languages accepted from sign-states are the same as those existing between zero-states.

Theorem 205. *Let q_1 and q_2 be sign-states. The following assertions are equivalent :*

1. *There exists a path from q_1 to q_2.*

2. $L_{\mathcal{A}}(q_1 \to q_1) \subseteq L_{\mathcal{A}}(q_2 \to q_2)$.

3. $L_{\mathcal{A}}(q_1) \subseteq L_{\mathcal{A}}(q_2)$.

Proof. The proof is similar to the proof of Theorem 163. □

7.3.2 Sign-SCCs

In this subsection, we characterize the SCCs having a sign-state, called sign-SCCs. The results presented are generalizations of those presented in Section 7.2.2. In particular, we show that there is a vector-space associated to each sign-SCC characterizing the sign-SCC and the concept of *representative matrix* also applies to sign-SCC.

Lemma 206. *In any maximal SCC of $\mathcal{A}$, there is at most one sign-state.*

Proof. Let q_1 and q_2 be sign-states in a maximal SCC $\mathcal{S}$. From Theorem 205, $L_{\mathcal{A}}(q_1) = L_{\mathcal{A}}(q_2)$ and since $\mathcal{A}$ is reduced minimal, $q_1 = q_2$. □

Definition 207. *A* sign-SCC *is a maximal strongly connected component having a sign-state.*

Definition 208. *Let $\mathcal{S}$ be a sign-SCC and let $q_{\mathtt{s}}$ be the sign-state of $\mathcal{S}$. The* dimension *of $\mathcal{S}$, written* $\dim(\mathcal{S})$*, is the dimension of the set* $\{\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}})\}$.

Example 209. *In the NDD $\mathcal{A}_{\mathtt{sign}}$ of Fig.7.2, there are four sign-SCCs. The dimension of the sign-SCC associated to q_1 (resp. q_4, q_5, q_6) is 0 (resp. 1, 2 and 1). Note that q_2 forms a SCC with no sign-state.*

Theorem 210. *Let $\mathcal{S}$ be a sign-SCC, let $q_{\mathtt{s}}$ be the sign-state of $\mathcal{S}$ and let $d = \dim(\mathcal{S})$.*

There exists an integer matrix $\mathbf{B}$ of rank $n - d$ such that for all states $q \in \mathcal{S}$ and words $v_q \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$,

- $\{\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q)\}$
$= \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{B}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} v_q \rangle_{r,n}) = \mathbf{0} \wedge \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$,

- $\mathrm{aff}_{\mathbb{Q}}(\{\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q)\})$
$= \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} v_q \rangle_{r,n}) = \mathbf{0}\}$.

Proof. The proof is similar to the proof of Theorem 172. □

Based on the previous theorem, we introduce the notion of representative matrix.

Definition 211. *A* representative matrix *of a sign-SCC $\mathcal{S}$ of dimension d is an integer matrix $\mathbf{B}$ of rank $n-d$ such that if $q_{\mathtt{s}}$ is the sign-state of $\mathcal{S}$, for all states $q \in \mathcal{S}$ and all $v_{q_{\mathtt{s}}q} \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q)$,*

- $\{\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q)\}$
$= \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{B}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} v_q \rangle_{r,n}) = \mathbf{0} \wedge \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$,
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q)\})$
$= \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} v_q \rangle_{r,n}) = \mathbf{0}\}$.

Corollary 212. *For each sign-SCC $\mathcal{S}$, there exists a representative matrix.*

Proof. Direct consequence of Theorem 210 and of the definition of a representative matrix. □

With minor changes, the procedure GETREPRESENTATIVEMATRIX applies to sign-SCC and we have the following Theorem.

Theorem 213. *Let $\mathcal{S}$ be a sign-SCC, let $q_{\mathtt{s}}$ be the sign-state of $\mathcal{S}$ and let $Q_{\mathcal{S}} \subseteq Q$ be the set of states in $\mathcal{S}$.*

There exists an algorithm GETREPRESENTATIVEMATRIX *which, given the reduced minimal NDD $\mathcal{A}$ and the sign-SCC $\mathcal{S}$, computes a representative matrix of $\mathcal{S}$ and whose time complexity is $\mathcal{O}(|Q_{\mathcal{S}}| \cdot |\Sigma_r^n| \cdot n^2)$.* □

We conclude this section by characterizing the incoming transitions in states of a sign-SCC.

Theorem 214. *Let $\mathcal{S}$ be a sign-SCC and let $q_{\mathtt{s}}$ be the sign-state of $\mathcal{S}$. Let $d = \dim(\mathcal{S})$.*

Each state of $\mathcal{S}$ has r^d incoming transitions from states in $\mathcal{S}$.

Proof. The proof is similar to the proof of Theorem 177. □

7.3.3 Sign-SCCs and Faces of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$

In this section, we emphasize the relationship between sign-SCCs of the NDD $\mathcal{A}$ representing $P \cap \mathbb{Z}^n$ and the faces $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C$ where C is the characteristic cone of P. The technical proofs are similar to those presented in Section 7.2.3, the required modifications having already been introduced in the preceding section.

We first present results on the properties of faces of the polyhedron $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+C$. Recall that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$, and no inequation in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$ is redundant in $\mathbf{C}\mathbf{x} \leq \mathbf{0}$.

Lemma 215. $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}(\mathbf{x} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = 0\}$.

Proof. Direct consequence of Lemma28. □

Lemma 216. *Let F be a facet of C, with $F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\}$ for some inequation $\mathbf{c}.\mathbf{x} \leq 0$ in $\mathbf{C}^+\mathbf{x} \leq \mathbf{0}$.*

$$\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}^{=}(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = \mathbf{0} \wedge \mathbf{c}.(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = 0\}.$$

Proof. Direct consequence of Lemma 29. □

Lemma 217. *For each face F of C,* $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F) = \mathrm{aff}_{\mathbb{Q}}\big(\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\big)$.

Proof. See Section 7.8.5. □

Lemma 218. *Let $q_{\mathtt{s}}$ be a sign-state and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{s}}$. Let $F = \{\mathbf{x} \in C \mid \bigwedge_{i\in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}$.*

- *For all $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{s}})$, $\langle u\rangle_{r,n} \in \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$,*
- *There exists a word $u \in (\Sigma_r^n)^*$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{s}})$ such that $\langle u\rangle_{r,n} \in \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ and for all proper faces F' of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$, $\langle u\rangle_{r,n} \notin F'$,*
- $L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}}) = \{v \in (\Sigma_r^n)^* \mid \langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)\}$,
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}})\}) = \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)$.

Proof. The proof is similar to the proof of Lemma 179. □

Thanks to the previous lemma, we deduce the following theorem, and this constitutes the first association between zero-SCCs and faces of C.

Theorem 219. *For each sign-SCC $\mathcal{S}$, one can associate one and only one face $F_{\mathcal{S}}$ of C such that*

$$\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F_{\mathcal{S}}) = \mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}})\}),$$

where $q_{\mathtt{s}}$ is the sign-state of $\mathcal{S}$.

Proof. This is a direct consequence of Lemma 218 and of the fact that for all faces F_1, $_2$ of a cone, $\mathrm{lin}_{\mathbb{Q}}(F_1) = \mathrm{lin}_{\mathbb{Q}}(F_2)$ iff $F_1 = F_2$, as proved in Lemma 30. □

Remark 220. *Given a sign-SCC $\mathcal{S}$ and a representative matrix $\mathbf{B}$ of $\mathcal{S}$, if $F_{\mathcal{S}}$ is such that $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) = \mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{s}} \to q_{\mathtt{s}})\})$ where $q_{\mathtt{s}}$ is the sign-state of $\mathcal{S}$, then, by definition of a representative matrix, we have*

- $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F_{\mathcal{S}}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) = \mathbf{0}\}$, *and*
- $\mathrm{lin}_{\mathbb{Q}}(F_{\mathcal{S}}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$. □

Lemma 221. *For each face F of the characteristic cone C, there exists one and only one sign-state, denoted q_F, such that the encodings, possibly suffixed by a sequence of $\alpha_{\mathtt{sign}}$ symbols, of all integer elements in $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ which do not belong to a proper face of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ label paths from $q_{\mathtt{I}}$ to q_F.*

In addition, the state q_F is such that

- $L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)\}$, *and*
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) \supseteq \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)$.

Proof. See Section 7.8.6. □

Theorem 222. *For each face F of C, there is one and only one sign-SCC $\mathcal{S}_F$ such that if $q_{\mathtt{s}}$ is the sign-state of $\mathcal{S}_F$, the following assertions hold.*

- *There is at least one encoding of an element of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ which labels a path from $q_{\mathtt{I}}$ to $q_{\mathtt{s}}$, and*
- $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \subseteq \mathrm{lin}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\})$.

Proof. Thanks to Lemma 221, there is at least one sign-SCC for which the assertions hold. So, it suffices to show that this sign-SCC is unique, and this is achieved by showing that the sign-state of each sign-SCC satisfying the assertions is reachable from the initial state by a path labeled by v such that $\langle v\rangle_{r,n}$ is in $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$ and does not belong to any proper face of F. Once this is proved, the claim is then a direct consequence of Lemma 221.

Let $q'_{\mathtt{s}}$ be the sign-state of a sign-SCC $\mathcal{S}'$ such that

- there exists an encoding $v \in (\Sigma_r^n)^+ \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q'_{\mathtt{s}})$ with $\langle v\rangle_{r,n} \in \langle\alpha_{\mathtt{sign}}\rangle_{r,n}+ F$, and
- $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \subseteq \mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}u\rangle_{r,n} \mid u \in L_{\mathcal{A}}(q'_{\mathtt{s}} \to q'_{\mathtt{s}})\})$.

Let $\alpha_{\mathtt{sign}}u'$ be the encoding of an element of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$ which does not belong to any proper face of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$. By hypothesis, we have $u'\in L_{\mathcal{A}}(q'_{\mathtt{s}}\to q'_{\mathtt{s}})$, and therefore $vu'\in L_{\mathcal{A}}(q_{\mathtt{I}}\to q'_{\mathtt{s}})$. Also, let $\mathbf{A}'\mathbf{x}\leq\mathbf{0}$ be the largest subsystem of inequations of $\mathbf{A}\mathbf{x}\leq\mathbf{0}$ such that for all $\mathbf{x}\in\mathbb{Q}^n$, $\mathbf{x}\in F\Rightarrow\mathbf{A}'\mathbf{x}=\mathbf{0}$. By definition, $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F=\{\mathbf{x}\in\mathbb{Q}^n\mid\mathbf{A}(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})\leq\mathbf{0}\wedge\mathbf{A}'(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})=\mathbf{0}\}$ and for each proper face F' of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$, there is at least one inequation $\mathbf{a}.\mathbf{x}\leq 0$ from the system $\mathbf{A}\mathbf{x}\leq\mathbf{0}$ but not in $\mathbf{A}'\mathbf{x}\leq\mathbf{0}$ such that for all $\mathbf{x}\in\mathbb{Q}^n$, $\mathbf{x}\in F'\Rightarrow\mathbf{a}.(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})=0$. Since $\langle v\rangle_{r,n},\langle\alpha_{\mathtt{sign}}u'\rangle_{r,n}\in\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$ and $\langle vu'\rangle_{r,n}=r^{|u'|}(\langle v\rangle_{r,n}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})+\langle\alpha_{\mathtt{sign}}u'\rangle_{r,n}$, $\langle vu'\rangle_{r,n}$ is in $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$. Also, by hypothesis, for each inequation $\mathbf{a}.\mathbf{x}\leq 0$ from the system $\mathbf{A}\mathbf{x}\leq\mathbf{0}$ but not in $\mathbf{A}'\mathbf{x}\leq\mathbf{0}$, $\mathbf{a}.(\langle v\rangle_{r,n}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})\leq 0$ and $\mathbf{a}.(\langle\alpha_{\mathtt{sign}}u'\rangle_{r,n}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})<0$, and therefore $\langle vu'\rangle_{r,n}$ does not belong to any proper face of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F$. □

Remark 223. *For any sign-SCC $\mathcal{S}$ and face F of C, if $\mathbf{B}$ is a representative matrix of $\mathcal{S}$ and if* $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F)\subseteq\mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}u\rangle_{r,n}\mid u\in L_{\mathcal{A}}(q_{\mathtt{s}}\to q_{\mathtt{s}})\})$ *where $q_{\mathtt{s}}$ is the sign-state of $\mathcal{S}$, then, by definition of a representative matrix,*

$$\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F)\subseteq\{\mathbf{x}\in\mathbb{Q}^n\mid\mathbf{B}(\mathbf{x}-\langle\alpha_{\mathtt{sign}}\rangle_{r,n})=\mathbf{0}\}.$$

□

Combining the fact that each sign-SCC $\mathcal{S}$ is associated to one and only one face $F_{\mathcal{S}}$ of C as detailed in Theorem 219, and conversely, that each face F of C is associated to one and only one sign-SCC $\mathcal{S}_F$ as detailed in Theorem 222, we deduce the following theorems regarding the linear hulls of C and of the facets of C. The motivation behind those theorems is to be able to recover from the representative matrices of the zero-SCCs of dimension $\dim(C)$ and $\dim(C)-1$ a set of inequations $\mathbf{C}'\mathbf{x}\leq\mathbf{0}$ such that the system $\mathbf{C}'\mathbf{x}\leq\mathbf{0}$ is equivalent to $\mathbf{C}\mathbf{x}\leq\mathbf{0}$. The algorithm is given in the next section.

Theorem 224. *For each sign-SCC $\mathcal{S}$, the dimension of $\mathcal{S}$ is at most* $\dim(C)$.

Proof. Direct consequence of the definition of a sign-SCC and of Theorem 219. □

Theorem 225. *There is one and only one sign-SCC $\mathcal{S}$ of dimension* $\dim(C)$ *and if $\mathbf{B}$ is a representative matrix of $\mathcal{S}$, we have*

$$\{\mathbf{x}\in\mathbb{Q}^n\mid\mathbf{B}\mathbf{x}=\mathbf{0}\}=\mathrm{lin}_{\mathbb{Q}}(C).$$

Proof. The proof is similar to the proof of Theorem 186. □

Theorem 226. *Let* $\mathbf{c}.\mathbf{x} \leq 0$ *be an inequation in* $\mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$ *and let* F *be the facet of* C *such that* $F = \{\mathbf{x} \in C \mid \mathbf{c}.\mathbf{x} = 0\}$*. At least one of the following assertions holds.*

- *There exists a sign-SCC* $\mathcal{S}$ *whose dimension is* $\dim(C) - 1$ *and such that for any representative matrix* $\mathbf{B}$ *of* $\mathcal{S}$*,* $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(F)$*.*
- *For all* $\mathbf{x} \in \mathbb{Q}^n$*, we have*

$$\left(\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \bigwedge_{\alpha_{\mathtt{sign}}[i]=0} \mathbf{x}[i] \geq 0 \wedge \bigwedge_{\alpha_{\mathtt{sign}}[i]=r-1} \mathbf{x}[i] \leq 0\right) \Rightarrow \mathbf{c}.\mathbf{x} \leq 0.$$

Proof. Thanks to Theorem 219, there exists one sign-SCC $\mathcal{S}$ such that if $q_{\mathtt{s}}$ is its sign-state and $\mathbf{B}$ is a representative matrix of $\mathcal{S}$, we have

- $\mathrm{lin}_{\mathbb{Q}}(F) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$, and
- there is an encoding $v \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{s}})$ such that $\langle v \rangle_{r,n} \in \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F$.

If $\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$, then, the first assertion holds.

Suppose on the other hand that $\mathrm{lin}_{\mathbb{Q}}(F) \subset \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}$. Since $\dim(F) = \dim(C) - 1$, by definition of the dimension of a sign-SCC and thanks to Theorem 224, the dimension of $\mathcal{S}$ is $\dim(C)$. Thanks to Theorem 225,

$$\mathrm{lin}_{\mathbb{Q}}(C) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\}. \tag{7.42}$$

We prove by contradiction that for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\left(\mathbf{C}^{=}\mathbf{x} = \mathbf{0} \wedge \bigwedge_{\alpha_{\mathtt{sign}}[i]=0} \mathbf{x}[i] \geq 0 \wedge \bigwedge_{\alpha_{\mathtt{sign}}[i]=r-1} \mathbf{x}[i] \leq 0\right) \Rightarrow \mathbf{c}.\mathbf{x} \leq 0.$$

Suppose that the assertion does not hold. There would exist a vector $\mathbf{y} \in \mathbb{Q}^n$ with $\bigwedge_{\alpha_{\mathtt{sign}}[i]=0} \mathbf{y}[i] \geq 0$, $\bigwedge_{\alpha_{\mathtt{sign}}[i]=r-1} \mathbf{y}[i] \leq 0$ and $\mathbf{C}^{=}\mathbf{y} = \mathbf{0}$ but $\mathbf{c}.\mathbf{y} > 0$. Since each component of $\mathbf{y}$ is a rational $\frac{n_i}{d_i}$, one could multiply $\mathbf{y}$ by the lowest common multiple of $d_1, \ldots, d_n$ and obtain a positive integer vector, and therefore, there exists a vector $\mathbf{y}' \in \mathbb{Z}^n$ such that $\bigwedge_{\alpha_{\mathtt{sign}}[i]=0} \mathbf{y}'[i] \geq 0$, $\bigwedge_{\alpha_{\mathtt{sign}}[i]=r-1} \mathbf{y}'[i] \leq 0$ and $\mathbf{C}^{=}\mathbf{y}' = \mathbf{0}$ but $\mathbf{c}.\mathbf{y}' > 0$. Let $\mathbf{z} = \mathbf{y}' - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$. By definition, $\bigwedge_{\alpha_{\mathtt{sign}}[i]=0} \mathbf{y}'[i] \geq 0 \wedge \bigwedge_{\alpha_{\mathtt{sign}}[i]=r-1} \mathbf{y}'[i] \leq 0$ is equivalent to $\mathrm{sign}_r(\mathbf{z}) = \alpha_{\mathtt{sign}}$, and thus we have

$$\mathrm{sign}_r(\mathbf{z}) = \alpha_{\mathtt{sign}} \wedge \mathbf{C}^{=}(\mathbf{z} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = \mathbf{0} \wedge \mathbf{c}.(\mathbf{z} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) > 0 \tag{7.43}$$

So, there exists a word $u \in (\Sigma_r^n)^*$ such that

$$\mathbf{C}^{=}(\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = \mathbf{0} \wedge \mathbf{c}.(\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) > 0 \quad (7.44)$$

Given (7.42), by definition of a representative matrix, $u \in L_{\mathcal{A}}(q_{\mathtt{s}} \rightarrow q_{\mathtt{s}})$. Recall that by hypothesis, $v \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{s}})$ and $\langle v \rangle_{r,n} \in \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F$. Therefore, for all $k \in \mathbb{N}$ we have $vu^k \in L_{\mathcal{A}}(q_{\mathtt{I}} \rightarrow q_{\mathtt{s}})$. Also we have $\mathbf{c}.(\langle v \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = 0$. Let $w \in (\Sigma_r^n)^*$ be a word labeling a path from $q_{\mathtt{s}}$ to an accepting state of $\mathcal{A}$. We have

$$\begin{aligned}
\mathbf{c}.\langle vu \rangle_{r,n} &= \mathbf{c}.(r^{|v|}\langle v \rangle_{r,n} + \langle ou \rangle_{r,n}) \\
&= r^{|u|}\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \mathbf{c}.(\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} - r^{|u|}\langle \alpha_{\mathtt{sign}} \rangle_{r,n}) \\
&= \mathbf{c}.\langle \alpha_{\mathtt{sign}} u \rangle_{r,n} \\
&> \mathbf{c}.\langle \alpha_{\mathtt{sign}} \rangle_{r,n} \\
&> \mathbf{c}.\langle v \rangle_{r,n}
\end{aligned}$$

So, thanks to Lemma 85, for any b, $\exists k_b \in \mathbb{N}$ such that $\mathbf{c}.\langle vu^{k_b} w \rangle_{r,n} > b$, with $\langle vu^{k_b} w \rangle_{r,n} \in P$. But this leads to a contradiction. Indeed, from Theorem 21, $P = Q + C$ for some polytope Q, and therefore, $\exists b_{\max} \in \mathbb{Q}$ such that for all $\mathbf{x} \in P$, $\mathbf{c}.\mathbf{x} \leq b_{\max}$. □

Theorem 227. *Let $\mathcal{S}$ be a sign-SCC whose dimension is $\dim(C) - 1$ and let $\mathbf{B}$ be a representative matrix of $\mathcal{S}$.*

There exists one and only one facet F such that

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(F).$$

Proof. Thanks to Theorem 219, there exists one and only one face F of C such that

$$\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}\mathbf{x} = \mathbf{0}\} = \mathrm{lin}_{\mathbb{Q}}(F).$$

So, by definition $\dim(F) = \dim(C) - 1$ and, thanks to Theorem 27, F is a facet. □

Theorem 228. *Let $\mathcal{S}$ be a sign-SCC with $\dim(\mathcal{S}) = \dim(C) - 1$, and let $q_{\mathtt{s}}$ be its sign-state and $\mathbf{B}$ be a representative matrix of $\mathcal{S}$.*

There is one and only one inequation $\mathbf{c}.\mathbf{x} \leq 0 \in \mathbf{C}^{+}\mathbf{x} \leq \mathbf{0}$ such that

1. *Each row of $\mathbf{B}$ is a linear combination of the rows of $\mathbf{C}^{=}$ and of $\mathbf{c}^{\mathtt{t}}$,*
2. *For all rows $\mathbf{a}^{\mathtt{t}}$ of $\mathbf{B}$ either $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \leq 0$ or $\mathbf{x} \in C \Rightarrow \mathbf{a}.\mathbf{x} \geq 0$.*

3. *There is a row* $\mathbf{a}^\intercal$ *of* $\mathbf{B}$ *such that* $\mathbf{a}^\intercal$ *is not a linear combination of the rows of* $\mathbf{C}^=$*,*

4. *For all rows* $\mathbf{a}^\intercal$ *of* $\mathbf{B}$ *such that* $\mathbf{a}^\intercal$ *is not a linear combination of the rows of* $\mathbf{C}^=$*, either for all* $\mathbf{x} \in \mathbb{Q}^n$*,* $\mathbf{C}^=\mathbf{x} = \mathbf{0} \wedge \mathbf{a}.\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^=\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} \leq 0$*, or for all* $\mathbf{x} \in \mathbb{Q}^n$*,* $\mathbf{C}^=\mathbf{x} = \mathbf{0} \wedge (-\mathbf{a}).\mathbf{x} \leq 0 \Leftrightarrow \mathbf{C}^=\mathbf{x} = \mathbf{0} \wedge \mathbf{c}.\mathbf{x} \leq 0$*.*

Proof. The proof is similar to the proof of Theorem 189. □

7.3.4 Algorithm

In this section, we present an algorithm that, given reduced minimal NDD $\mathcal{A}$ representing the integer elements of a polyhedron $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ such that $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}\}$, synthesizes a system of linear inequations corresponding to the characteristic cone C of P. The algorithm uses the properties given in Section 7.3.3 regarding the sign-SCCs of dimension $d_{\max}$ and $d_{\max} - 1$ where $d_{\max}$ is the dimension of the characteristic cone, and its overall time complexity is polynomial with respect to the size of the input NDD.

The algorithm and the justification of its correctness are similar to what has been done in Section 7.2.4. So, we simply give the formal description of the algorithm in Figure 7.7 and state its correctness. Note that in this case, one needs to identify the sign symbol $\alpha_{\texttt{sign}}$ by checking the label of the transition outgoing from the initial state.

Theorem 229. *Let* $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\texttt{I}}, Q_{\texttt{F}})$ *be the reduced minimal NDD representing the integer elements of a polyhedron* $P \subseteq \mathbb{Q}^n$ *with* $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}\}$*.*

Let $\mathbf{C}\mathbf{x} \leq \mathbf{0}$ *be the system of linear inequations returned by the function* CHARCONEFORMULA*, described in Fig.7.7.*

Then, $\mathbf{C}\mathbf{x} \leq \mathbf{0}$ *is such that*

$$\text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}.$$

The time complexity of CHARCONEFORMULA *is* $\mathcal{O}(|\delta| \cdot n^2 + m \cdot |Q| \cdot n)$ *where* m *is the number of sign-SCC of dimension* $\dim(\text{char-cone}(P)) - 1$*.*

Proof. The proof is similar to the proof of Theorem 190. □

function CHARCONEFORMULA(NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$) : system of linear inequations

```
 1:  var S_SCC : set of sign-SCC;
 2:      Q_visited : set of state;
 3:      d_max : integer;
 4:      α_sign : symbol;
 5:      B,C : array of rational;
 6:  procedure EXPLORE-BW(state q, word w)
 7:      var q' : state;
 8:          α : symbol;
 9:          S : sign-SCC;
10:          a : vector of rational;
11:      begin
12:          Q_visited := Q_visited ∪ {q};
13:          if q is a sign-state then
14:              begin
15:                  let S ∈ S_SCC such that q is sign-state of S;
16:                  if GETDIMENSIONSCC(S) = d_max − 1 then
17:                      begin
18:                          B := GETREPRESENTATIVEMATRIX(S);
19:                          let a such that a^t is a row of B
                                and a.(⟨α_sign w⟩_{r,n} − ⟨α_sign⟩_{r,n}) ≠ 0;
20:                          if a.(⟨α_sign w⟩_{r,n} − ⟨α_sign⟩_{r,n}) > 0 then
21:                              a := −a;
22:                          C := [ C ; a^t ];
23:                      end
24:              end
25:          for each α ∈ Σ_r^n, q' ∈ Q \ Q_visited such that δ(q', α) = q do
26:              EXPLORE-BW(q', αw);
27:      end

    (...)
```

Figure 7.7: Function CHARCONEFORMULA

(...)

28: **begin**
29: **let** $\alpha_{\mathtt{sign}}$ **such that** $\delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) \in Q$;
30: S_{SCC} := GETSIGNSCCS($\mathcal{A}$, $\alpha_{\mathtt{sign}}$);
31: $d_{\max}$:= $\max_{\mathcal{S} \in S_{SCC}}$(GETDIMENSIONSCC ($\mathcal{S}$);
32: **let** $\mathcal{S}_{\max}$ **such that** GETDIMENSIONSCC($\mathcal{S}_{\max}$) = $d_{\max}$;
33: $\mathbf{B}$:= GETREPRESENTATIVEMATRIX($\mathcal{S}_{\max}$);
34: $\mathbf{C}$:= $\mathbf{I}_n$;
35: **for each** $i \in \{1, \ldots n\}$ **do**
36: **if** ($\alpha_{\mathtt{sign}}[i] = 0$) **then** $\mathbf{C}[i, i]$:= -1;
37: $\mathbf{C} := \begin{bmatrix} \mathbf{C} \\ \mathbf{B} \\ -\mathbf{B} \end{bmatrix}$;
38: EXPLORE-BW(GETSIGNSTATE($\mathcal{S}_{\max}$), ε);
39: **return** $\mathbf{Cx} \leq \mathbf{0}$;
40: **end**

Figure 7.8: Function CHARCONEFORMULA (continued)

Example 230. *In the NDD $\mathcal{A}_{\mathtt{sign}}$ displayed in Fig.7.6, the sign-SCC with the maximal dimension is the one associated to the state q_5. Its dimension is 2 and the associated vector space is $\mathbb{Q}^2$. A representative matrix is $\begin{bmatrix} 0 & 0 \end{bmatrix}$. There are two sign-SCCs with dimension 1, those associated to the sign-states q_4 and q_6. The associated vector spaces are $\{(x,y) \in \mathbb{Q}^2 \mid x - 2 \cdot y = 0\}$ and $\{(x,y) \in \mathbb{Q}^2 \mid x - y = 0\}$ respectively. A representative matrice for the sign-SCC associated to q_4 is $\begin{bmatrix} 1 & -2 \end{bmatrix}$ and a representative matrix for the sign-SCC associated to q_6 is $\begin{bmatrix} 1 & -1 \end{bmatrix}$. Given the labels of the paths from q_4 and q_6 to q_5, the characteristic cone is $\{(x,y) \in \mathbb{Q}^2 \mid x \leq 0 \wedge y \leq 0 \wedge x - y \leq 0 \wedge -x + 2 \cdot y \leq 0\}$.* □

7.4 Synthesis of Basis of $P \cap \mathbb{Z}^n$

In this section, we present an algorithm that, given a reduced minimal NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ representing the integer elements of a polyhedron $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ such that $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$ and an integer matrix $\mathbf{C}$ such that $\text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$, generates the basis of $P \cap \mathbb{Z}^n$. The set C will denote the characteristic cone of P, i.e. $C = \text{char-cone}\, P$.

Remark 231. *Thanks to Lemma 194, C is pointed, and therefore P is also pointed.* □

Note that since a deterministic NDD $\mathcal{A}_C$ accepting the encodings of the integer elements in C can be generated in linear time from $\mathcal{A}$ and $|\mathcal{A}_C| \leq |\mathcal{A}|$, the complexity of procedures presented in Section 7.1 computing the constants and the periods of the Hilbert basis must be revisited. Indeed, given $\mathcal{A}_C$, the worst-case time complexities of the procedures are $\mathcal{O}(2^{|Q|^2 \cdot 2^n})$. The procedures given in this section have also an exponential worst-case complexities. However, they are much more efficient in practice as already discussed in Section 7.1.

In Section 7.4.1, we express the sets $P \cap \mathbb{Z}^n$ and $C \cap \mathbb{Z}^n$ as a finite union of sets $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$, where $S_{\mathcal{A}}^q$ is the set of vectors whose encodings label paths from $q_{\mathtt{I}}$ to q in $\mathcal{A}$. Then, in Section 7.4.2, we show that for each $q \in Q$, there exists an extended Hilbert basis generating the set $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$, and in Section 7.4.3, we show that an Hilbert basis generating $C \cap \mathbb{Z}^n$ and an extended Hilbert basis generating $P \cap \mathbb{Z}^n$ can be computed from the extended Hilbert basis generating $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$. Finally, the algorithms performing the computation of the extended Hilbert basis of $P \cap \mathbb{Z}^n$ is given in Section 7.4.4.

7.4.1 Association of Sets of Vectors to States

Recall that for each $q \in Q$, the set $S^q_{\mathcal{A}}$ is the set of vectors whose encodings label paths from $q_{\mathtt{I}}$ to q, i.e. $S^q_{\mathcal{A}} = \{\langle u \rangle_{r,n} \mid u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)\}$.

Remark 232. *Since $\mathcal{A}$ is reduced minimal, if $q \neq q'$, then $S^q_{\mathcal{A}} \cap S^{q'}_{\mathcal{A}} = \emptyset$.* □

By definition, the set $P \cap \mathbb{Z}^n$ can be expressed in terms of $S^q_{\mathcal{A}}$. Indeed, it suffices to take the union of the sets $S^q_{\mathcal{A}}$ for all $q \in Q_{\mathtt{F}}$. Since P is a polyhedron and $\mathcal{A}$ is reduced, the vectors in $C \cap \mathbb{Z}^n$ can be added to the vectors in any $S^q_{\mathcal{A}}$.

Theorem 233. $P \cap \mathbb{Z}^n = \cup_{q \in Q_{\mathtt{F}}} S^q_{\mathcal{A}} = \cup_{q \in Q_{\mathtt{F}}} \big(S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)\big)$.

Proof. By definition, $P \cap \mathbb{Z}^n = \{\langle u \rangle_{r,n} \mid \hat{\delta}(q_{\mathtt{I}}, u) \in Q_{\mathtt{F}}\}$. Therefore, by definition of $S^q_{\mathcal{A}}$,

$$P \cap \mathbb{Z}^n = \cup_{q \in Q_{\mathtt{F}}} S^q_{\mathcal{A}} \tag{7.45}$$

In addition, for all $\mathbf{x} \in P \cap \mathbb{Z}^n$ and for all $\mathbf{y} \in C \cap \mathbb{Z}^n$, $\mathbf{x} + \mathbf{y} \in P \cap \mathbb{Z}^n$. So, $(P \cap \mathbb{Z}^n) + (C \cap \mathbb{Z}^n) \subseteq P \cap \mathbb{Z}^n$. Since $\mathbf{0} \in (C \cap \mathbb{Z}^n)$, we have

$$P \cap \mathbb{Z}^n = P \cap \mathbb{Z}^n + C \cap \mathbb{Z}^n = \cup_{q \in Q_{\mathtt{F}}} \big(S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)\big). \tag{7.46}$$

□

Thanks to Theorem 199, the integer elements in C can also be deduced from the sets $S^q_{\mathcal{A}}$.

Theorem 234. *Let Q_C be the set of states q such that there is a path from q to a sign-state labeled by a sequence of $\alpha_{\mathtt{sign}}$ symbols.*

$$C \cap \mathbb{Z}^n = -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (\cup_{q \in Q_C} S^q_{\mathcal{A}}) = -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(\cup_{q \in Q_C} \big(S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)\big)\right).$$

Proof. Thanks to Theorem 199, we have

$$\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C \cap \mathbb{Z}^n = \cup_{q \in Q_C} S^q_{\mathcal{A}} \tag{7.47}$$

Also, since C is a cone, by definition, we have

$$C \cap \mathbb{Z}^n = (C \cap \mathbb{Z}^n) + (C \cap \mathbb{Z}^n). \tag{7.48}$$

Combining (7.47) and (7.48), we have

$$\begin{aligned} C \cap \mathbb{Z}^n &= \cup_{q \in Q_C} (-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + S^q_{\mathcal{A}}) \\ &= \cup_{q \in Q_C} \big((-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + S^q_{\mathcal{A}}) + (C \cap \mathbb{Z}^n)\big) \\ &= -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (\cup_{q \in Q_C} (S^q_{\mathcal{A}}) + (C \cap \mathbb{Z}^n)). \end{aligned}$$

□

7.4.2 Extended Hilbert Basis for States in $\mathcal{A}$

In this section, we show that for each $q \in Q$, there exists an extended Hilbert basis (X, Y) with $X + \mathrm{cone}_{\mathbb{Z}}(Y) = S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$ such that the minimal encoding of each $\mathbf{x} \in X$ labels an acyclic path from $q_{\mathtt{I}}$ to q and Y is an Hilbert basis generating $C \cap \mathbb{Z}^n$.

Since $\mathcal{A}$ is a reduced minimal NDD representing the integer elements of P, for each inequation defining P there is a bound on the value of the left-hand side when replacing in the inequation the vector of indeterminates by any vector whose encodings label paths in $\mathcal{A}$, as proved in the following lemma.

Lemma 235. *For each inequation $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, there exists an upper bound $b_{\max} \in \mathbb{Z}$ such that for all encodings u labeling a path in $\mathcal{A}$, $\mathbf{a}.\langle u \rangle_{r,n} \leq b_{\max}$.*

Proof. Let $q \in Q$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$. Since $\mathcal{A}$ is reduced minimal, there exists a word w with $|w| \leq |Q|$ labeling a path from q to an accepting state. By definition, $\mathbf{a}.\langle uw \rangle_{r,n} \leq b$. Thanks to Lemma 81, $\mathbf{a}.\langle uw \rangle_{r,n} = r^{|w|}\mathbf{a}.\langle u \rangle_{r,n} + \mathbf{a}.\langle ow \rangle_{r,n}$ and by definition of the encoding scheme, $\mathbf{a}.\langle ow \rangle_{r,n} \geq (1 - r^{|w|+1}) \| \mathbf{a}^- \|$. So, we deduce that

$$\begin{aligned} \mathbf{a}.\langle u \rangle_{r,n} &= r^{-|w|} \left(\mathbf{a}.\langle uw \rangle_{r,n} - \mathbf{a}.\langle ow \rangle_{r,n} \right) \\ &\leq r^{-|w|} \left(b - (1 - r^{|w|+1}) \| \mathbf{a}^- \| \right) \\ &\leq b + r \| \mathbf{a}^- \|. \end{aligned}$$

□

As a corollary of the above lemma, we deduce that for each inequation defining P, the left-hand side can not increase when considering the vectors corresponding to a path and the same path with a loop, as shown in the following lemma.

Lemma 236. *For each inequation $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$, for each state $q \in Q$ and for each $u, v \in (\Sigma_r^n)^*$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ and $v \in L_{\mathcal{A}}(q \to q)$, we have*

$$\mathbf{a}.\langle uv \rangle_{r,n} \leq \mathbf{a}.\langle u \rangle_{r,n}.$$

Proof. The proof is by contradiction. Suppose that $q \in Q$ and $u, v \in (\Sigma_r^n)^*$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q)$ and $v \in L_{\mathcal{A}}(q \to q)$, and $\mathbf{a}.\langle uv \rangle_{r,n} > \mathbf{a}.\langle u \rangle_{r,n}$. So, thanks to Lemma 85, for all b, there exists k_b such that $\mathbf{a}.\langle uv^{k_b} \rangle_{r,n} > b$, contradicting Lemma 235. □

From the above lemma, we deduce that the difference between two vectors whose encodings label paths differing only by a loop is in $C \cap \mathbb{Z}^n$.

Lemma 237. *Let $q_1, q_2 \in Q$. If $u, v, w \in (\Sigma_r^n)^*$ are such that $\hat{\delta}(q_\mathtt{I}, u) = q_1$, $\hat{\delta}(q_\mathtt{I}, uv) = q_1$ and $\hat{\delta}(q_\mathtt{I}, uvw) = q_2$, then*

$$\langle uvw \rangle_{r,n} - \langle uw \rangle_{r,n} \in C \cap \mathbb{Z}^n.$$

Proof. By definition, $C = \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0}\}$, and from Lemma 236, we have

$$\mathbf{A}\langle uv \rangle_{r,n} \leq \mathbf{A}\langle u \rangle_{r,n}.$$

So, thanks to Lemma 81, we have $\mathbf{A}\langle uvw \rangle_{r,n} \leq \mathbf{A}\langle uw \rangle_{r,n}$. □

As a direct consequence, we have the following lemma.

Lemma 238. *Let $q \in Q$ and $\mathbf{z} \in S_\mathcal{A}^q$.*

If the minimum encoding of $\mathbf{z}$ does not label an acyclic path, then $\mathbf{z}$ can be decomposed into a vector $\mathbf{x} \in S_\mathcal{A}^q$ and a vector $\mathbf{y} \in C \cap \mathbb{Z}^n \setminus \{\mathbf{0}\}$ such that $\mathbf{z} = \mathbf{x} + \mathbf{y}$ and the minimal encoding u of $\mathbf{x}$ labels an acyclic path from $q_\mathtt{I}$ to q.

Proof. Let w be the minimal encoding of $\mathbf{z}$. Since $\mathcal{A}$ is reduced minimal, $\hat{\delta}(q_\mathtt{I}, w) = q$. By hypothesis, there exist $u_1, u_2, u_3 \in (\Sigma_r^n)^*$ with $u_2 \neq \varepsilon$ such that $u_1u_2u_3 = w$, $\hat{\delta}(q_\mathtt{I}, u_1) = \hat{\delta}(q_\mathtt{I}, u_1u_2)$ and $\hat{\delta}(q_\mathtt{I}, u_1u_2u_3) = q$. Thanks to Lemma 237, $\langle u_1u_2u_3 \rangle_{r,n} - \langle u_1u_3 \rangle_{r,n} \in C \cap \mathbb{Z}^n$, i.e. $\mathbf{z} = \mathbf{z}_1 + \mathbf{y}_1$ with $\mathbf{z}_1 = \langle u_1u_3 \rangle_{r,n} \in S_\mathcal{A}^q$ and $\mathbf{y}_1 \in C \cap \mathbb{Z}^n$. Since w is the minimal encoding of $\mathbf{z}$, $\langle u_1u_2u_3 \rangle_{r,n} \neq \langle u_1u_3 \rangle_{r,n}$, and by definition of the encoding scheme, for all $j \in \{1, \ldots, n\}$, $|\langle u_1u_2u_3 \rangle_{r,n}[j]| \geq |\langle u_1u_3 \rangle_{r,n}[j]|$, and therefore, we have

$$\sum_{j=1}^{n} |\mathbf{z}_1[j]| < \sum_{j=1}^{n} |\mathbf{z}[j]|.$$

We apply recursively the same reasoning to $\mathbf{z}_1$ until reaching a vector whose minimum encoding labels an acyclic path. So, we generate a sequence of vectors $\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_t \in S_\mathcal{A}^q$ such that $\mathbf{z}_0 = \mathbf{z}$ and for all $0 \leq k$, $\mathbf{z}_k = \mathbf{z}_{k+1} + \mathbf{y}_{k+1}$ with $\mathbf{y}_{k+1} \in C \cap \mathbb{Z}^n$. From above, we have $0 \leq \sum_{j=1}^{n} |\mathbf{z}_{k+1}[j]| < \sum_{j=1}^{n} |\mathbf{z}_k[j]|$ for all k, and therefore, the sequence $\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_t$ is finite. By construction, $\mathbf{z}_0 = \mathbf{z}_t + \sum_{i=1}^{t} \mathbf{y}_i$, and by definition of a cone, $\sum_{i=1}^{t} \mathbf{y}_i \in C \cap \mathbb{Z}^n$. We conclude that $\mathbf{z} = \mathbf{z}_t + \mathbf{y}$ with $\mathbf{y} \in C \cap \mathbb{Z}^n$ and $\mathbf{z}_t \in S_\mathcal{A}^q \setminus \{\mathbf{z}\}$ such that the minimal encoding of $\mathbf{z}_t$ labels an acyclic path from $q_\mathtt{I}$ to q. □

Remark 239. *First note that given Lemma 71, in a reduced minimal NDD, two encodings labeling acyclic paths are encodings of different vectors. Also, in some cases, there are exactly as many constants in the minimal extended Hilbert basis as there are acyclic paths. This happens in particular when the characteristic cone of P is $\{\mathbf{0}\}$, which occurs if P is a polytope. In this case, $P \cap \mathbb{Z}^n$ is finite and each acyclic path from $q_{\mathtt{I}}$ to an accepting state q corresponds to a unique vector which can not be decomposed, and therefore, is a constant of the minimal extended Hilbert basis generating $P \cap \mathbb{Z}^n$.* □

Theorem 240. *Let $Y_{\min}$ be the minimal Hilbert basis of $C \cap \mathbb{Z}^n$.*

For each $q \in Q$, there exists a finite set $X_{q,\min}$ such that

1. *for each $\mathbf{x} \in X_{q,\min}$ there exists an acyclic path from $q_{\mathtt{I}}$ to q labeled by the minimal encoding of $\mathbf{x}$,*
2. *$(X_{q,\min}, Y_{\min})$ is a minimal extended Hilbert basis,*
3. *$(X_{q,\min}, Y_{\min})$ generates $S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n$.*

Proof. Let $q \in Q$ and let $S^{q,\mathtt{acyclic}}_{\mathcal{A}}$ be the set of vectors $\mathbf{x}$ such that there exists an acyclic path from $q_{\mathtt{I}}$ to q labeled by u with $\langle u \rangle_{r,n} = \mathbf{x}$. By definition, $S^{q,\mathtt{acyclic}}_{\mathcal{A}}$ is finite and $(S^{q,\mathtt{acyclic}}_{\mathcal{A}}, Y_{\min})$ is an extended Hilbert basis. Also, since C is pointed and $\mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = C \cap \mathbb{Z}^n$, thanks to Lemma 40, there exists a set $X_{q,\min} \subseteq S^{q,\mathtt{acyclic}}_{\mathcal{A}}$ such that $(X_{q,\min}, Y_{\min})$ is a minimal extended Hilbert basis and

$$X_{q,\min} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = S^{q,\mathtt{acyclic}}_{\mathcal{A}} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}).$$

By construction, for all $\mathbf{x} \in X_{q,\min}$, there exists an acyclic path from $q_{\mathtt{I}}$ to q labeled by u with $\langle u \rangle_{r,n} = \mathbf{x}$. Also, since $\mathcal{A}$ is reduced minimal, u must be the minimal encoding of $\mathbf{x}$. Indeed, otherwise, if $u_{\min} \neq u$ was the minimal encoding of $\mathbf{x}$, thanks to Lemma 71, it would only differ by a repetition of the sign symbol, and since $\mathcal{A}$ is minimal reduced, for all sign symbols α and $k \in \mathbb{N}$, $\hat{\delta}(q_{\mathtt{I}}, \alpha^k) = \delta(q_{\mathtt{I}}, \alpha)$, and therefore, u would not label an acyclic path.

Finally, we show that $S^{q,\mathtt{acyclic}}_{\mathcal{A}} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n$. By definition, $\mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = C \cap \mathbb{Z}^n$ and $S^{q,\mathtt{acyclic}}_{\mathcal{A}} \subseteq S^q_{\mathcal{A}}$, and therefore, we have

$$S^{q,\mathtt{acyclic}}_{\mathcal{A}} + \mathrm{cone}_{\mathbb{Z}}(Y_{\min}) \subseteq S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n. \tag{7.49}$$

Also, thanks to Lemma 238, by definition, for all $\mathbf{z} \in S^q_{\mathcal{A}} \setminus S^{q,\mathtt{acyclic}}_{\mathcal{A}}$, $\mathbf{z} = \mathbf{x} + \mathbf{y}$ with $\mathbf{x} \in S^q_{\mathcal{A}} \setminus \{\mathbf{z}\}$ and $\mathbf{y} \in C \cap \mathbb{Z}^n$ such that the minimal encoding of $\mathbf{x}$ labels an acyclic path from $q_{\mathtt{I}}$ to q, i.e. $\mathbf{x} \in S^{q,\mathtt{acyclic}}_{\mathcal{A}}$. So, we have

$$S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n \subseteq S^{q,\mathtt{acyclic}}_{\mathcal{A}} + C \cap \mathbb{Z}^n. \tag{7.50}$$

Combining (7.49) and (7.50), we conclude that

$$S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n = S^{q,\mathtt{acyclic}}_{\mathcal{A}} + C \cap \mathbb{Z}^n.$$

□

Remark 241. *For all extended Hilbert basis (X_q, Y) generating the set $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$, thanks to Theorem 34 and Lemma 40, if $(X_{q,\min}, Y_{\min})$ is the minimal extended Hilbert basis generating $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$, then $X_{q,\min} \subseteq X_q$ and $Y_{\min} \subseteq Y$.* □

7.4.3 From Basis of $S^q_{\mathcal{A}} + C \cap \mathbb{Z}^n$ to Basis of $P \cap \mathbb{Z}^n$

In this section, we first show how to generate the constants X of an extended Hilbert basis generating $P \cap \mathbb{Z}^n$ from extended Hilbert basis of sets $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$. Then, with a similar method, we show how to generate an Hilbert basis Y generating the set $C \cap \mathbb{Z}^n$. By definition, the pair (X, Y) is then an extended Hilbert basis generating $P \cap \mathbb{Z}^n$.

Theorem 242. *For each $q \in Q$, let (X_q, Y_q) be an extended Hilbert basis generating $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$, and let Y be an Hilbert basis generating $C \cap \mathbb{Z}^n$.*

The pair $(\cup_{q \in Q_{\mathtt{F}}} X_q, Y)$ is an extended Hilbert basis generating $P \cap \mathbb{Z}^n$.

Proof. Thanks to Theorem233, we have

$$P \cap \mathbb{Z}^n = \cup_{q \in Q_{\mathtt{F}}} \big(S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n) \big). \tag{7.51}$$

By hypothesis, $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n) = X_q + \mathrm{cone}_{\mathbb{Z}}(Y)$, and therefore, we have

$$P \cap \mathbb{Z}^n = (\cup_{q \in Q_{\mathtt{F}}} X_q) + \mathrm{cone}_{\mathbb{Z}}(Y). \tag{7.52}$$

Since $\cup_{q \in Q_{\mathtt{F}}} X_q$ is finite, by definition, the pair $(\cup_{q \in Q_{\mathtt{F}}} X_q, Y)$ is an extended Hilbert basis generating $P \cap \mathbb{Z}^n$. □

Comparing Theorems 233 and 234, from the above theorem, we might expect that an Hilbert basis generating $C \cap \mathbb{Z}^n$ would be given by the set $-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \cup_{q \in Q_C} X_q$, where Q_C is defined as in Theorem 234 and X_q is the set of constants of an extended Hilbert basis generating $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$. However, this is in general not correct since $\cup_{q \in Q_C} X_q$ could simply be $\langle \alpha_{\mathtt{sign}} \rangle_{r,n}$. The problem arises since $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} \in S^q_{\mathcal{A}}$ for some $q \in Q_C$. Indeed, thanks to Theorem 199, we have

$$S^q_{\mathcal{A}} \subseteq \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + C.$$

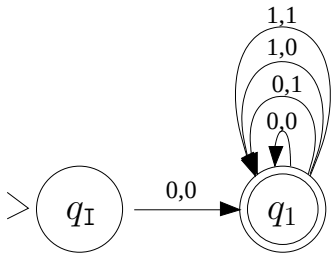


Figure 7.9: minimal reduced NDD representing $\{\mathbf{x} \in \mathbb{Z}^2 \mid \mathbf{x} \geq \mathbf{0}\}$.

Thus, for all $\mathbf{x} \in S^q_{\mathcal{A}} \setminus \{\langle \alpha_{\mathtt{sign}} \rangle_{r,n}\}$, we have

$$\mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n} \in C \cap \mathbb{Z}^n.$$

We deduce that $\mathbf{x}$ is not a constant of the minimal extended Hilbert basis generating $S^q_{\mathcal{A}} + C + \cap\mathbb{Z}^n$, although $\mathbf{x}$ might be an element in the minimal Hilbert basis of $C \cap \mathbb{Z}^n$. Consider for example the case where P is $\{\mathbf{x} \in \mathbb{Q}^2 \mid \mathbf{x} \geq \mathbf{0}\}$. The minimal NDD representing $P \cap \mathbb{Z}^2$ is given in Figure 7.9. Clearly, the set of constant in the minimal extended Hilbert basis of $S_{q_1} + C \cap \mathbb{Z}^n$ is $\{\mathbf{0}\}$.

Theorem 243. *For each $q \in Q$, let (X_q, Y_q) be an extended Hilbert basis generating $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$, and let Q_C be the set of states q such that there is a path from q to a sign-state labeled by a sequence of $\alpha_{\mathtt{sign}}$ symbols.*

Let $X'_{q_{\alpha_{\mathtt{sign}}}} = \left\{r \cdot \mathbf{x} + \langle o\alpha \rangle_{r,n} \mid \exists q \in Q, \alpha \in \Sigma^n_r \left(\delta(q, \alpha) = q_{\alpha_{\mathtt{sign}}} \wedge \mathbf{x} \in X_q\right)\right\}$ and let $q_{\alpha_{\mathtt{sign}}} \in Q$ such that $\delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = q_{\alpha_{\mathtt{sign}}}$.

The set $-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q\right)$ is an Hilbert basis generating $C \cap \mathbb{Z}^n$.

Proof. See Section 7.8.7. □

7.4.4 Algorithm

In this last section, we present an algorithm generating the basis of $P \cap \mathbb{Z}^n$ based on result given in Sections 7.4.2 and 7.4.3. The polyhedron P must be such that $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$, and consequently, P is pointed.

According to Theorems 242 and 243, the set of constants and the set of periods of an extended Hilbert basis generating $P \cap \mathbb{Z}^n$ can be computed from the sets of constants X_q of any extended Hilbert basis generating the sets $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$,

where C is the characteristic cone of P and $S^q_{\mathcal{A}}$ is the set of vectors whose encodings label paths from $q_{\mathtt{I}}$ to q. So, the main issue of this section is the computation of the sets X_q, with $q \in Q$.

Thanks to Theorem 240, for each $q \in Q$, if $(X_{q,\min}, Y_{\min})$ is the minimal extended Hilbert basis generating $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$, then each vector $\mathbf{x} \in X_{q,\min}$ has an encoding labeling an acyclic path from $q_{\mathtt{I}}$ to q. This translates into a first approach for computing the sets X_q. It suffices to compute for each q the set of vectors $S^{q,\mathtt{acyclic}}_{\mathcal{A}}$ having an encoding labeling an acyclic path from $q_{\mathtt{I}}$ to q. By definition, $S^{q,\mathtt{acyclic}}_{\mathcal{A}}$ is finite and from above,

$$X_{q,\min} \subseteq S^{q,\mathtt{acyclic}}_{\mathcal{A}}.$$

Therefore, one can choose $X_q = S^{q,\mathtt{acyclic}}_{\mathcal{A}}$. The drawback of this approach is that one systematically explores all acyclic paths in the NDD. In the following, we present an approach which filters out some of the acyclic paths. The property on which the filtering process relies is the following. If u and v label paths from $q_{\mathtt{I}}$ to q and $\langle u\rangle_{r,n} - \langle v\rangle_{r,n} \in (C \cap \mathbb{Z}^n) \setminus \{\mathbf{0}\}$, then $\langle u\rangle_{r,n}$ is not in $X_{q,\min}$ and for each w labeling a path from q to another state q', $\langle uw\rangle_{r,n}$ is not in $X_{q',\min}$. So, one does not need to explore the paths rooted at $q_{\mathtt{I}}$ and prefixed by u. Although the worst case complexity is identical to the complexity of the above algorithm, it significantly improves the performance in practice.

First, we give a formal proof of the correctness of the filtering criterion.

Lemma 244. *Let $q \in Q$ and $u, v \in (\Sigma^n_r)^*$ such that $\hat{\delta}(q_{\mathtt{I}}, u) = q = \hat{\delta}(q_{\mathtt{I}}, v)$.*

If $\langle u\rangle_{r,n} - \langle v\rangle_{r,n} \in (C \cap \mathbb{Z}^n) \setminus \{\mathbf{0}\}$, then for all $q' \in Q$ and $w \in (\Sigma^n_r)^*$ with $\hat{\delta}(q, w) = q'$, $\langle uw\rangle_{r,n}$ is not a constant of the minimal extended Hilbert basis generating $S^{q'}_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$.

Proof. By definition, C is a cone, and therefore, there exists a matrix $\mathbf{C}$ such that $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$. By hypothesis and by definition of $\mathbf{C}$, we have

$$\mathbf{C}(\langle u\rangle_{r,n} - \langle v\rangle_{r,n}) \leq \mathbf{0}. \tag{7.53}$$

We show that for all w, $\mathbf{C}(\langle uw\rangle_{r,n} - \langle vw\rangle_{r,n}) \leq \mathbf{0}$. Clearly, if $w = \varepsilon$, then $\mathbf{C}(\langle uw\rangle_{r,n} - \langle vw\rangle_{r,n}) \leq \mathbf{0}$. If $|w| \geq 1$, then, by definition of the encoding scheme, we have

$$\begin{aligned} \langle uw\rangle_{r,n} - \langle vw\rangle_{r,n} &= r^{|w|}\langle u\rangle_{r,n} + \langle ow\rangle_{r,n} - r^{|w|}\langle v\rangle_{r,n} - \langle ow\rangle_{r,n} \\ &= r^{|w|}(\langle u\rangle_{r,n} - \langle v\rangle_{r,n}). \end{aligned}$$

Therefore, $\langle uw \rangle_{r,n} - \langle vw \rangle_{r,n} \neq \mathbf{0}$ and $\mathbf{C}(\langle uw \rangle_{r,n} - \langle vw \rangle_{r,n}) \leq \mathbf{0}$.

So, we conclude that $\langle uw \rangle_{r,n} - \langle vw \rangle_{r,n} \in (C \cap \mathbb{Z}^n) \setminus \{\mathbf{0}\}$, i.e. $\langle uw \rangle_{r,n} = \langle vw \rangle_{r,n} + \mathbf{y}$ with $\langle uw \rangle_{r,n}, \langle vw \rangle_{r,n} \in S_{\mathcal{A}}^{q'}$ and $\mathbf{y} \in (C \cap \mathbb{Z}^n)$. So, by definition, $\langle uw \rangle_{r,n}$ is not a constant of the minimal extended Hilbert basis generating $S_{\mathcal{A}}^{q'} + (C \cap \mathbb{Z}^n)$. □

Remark 245. *Given a matrix* $\mathbf{C}$ *such that* $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$*, checking whether* $\mathbf{x} - \mathbf{y}$ *is in* $C \cap \mathbb{Z}^n$ *can be done by computing the product* $\mathbf{C}(\mathbf{x} - \mathbf{y})$ *in time proportion to* $\mathcal{O}(k \cdot n)$ *where* k *is the number of rows of* $\mathbf{C}$*.* □

Based on the above considerations, there exists an algorithm COMPUTEBASIS computing an extended Hilbert basis generating the set $P \cap \mathbb{Z}^n$ with $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}\}$ given the reduced minimal NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\texttt{I}}, Q_{\texttt{F}})$ representing $P \cap \mathbb{Z}^n$ and a matrix $\mathbf{C}$ such that the characteristic cone of P is $\{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$. The algorithm computes first the sets X_q for each state $q \in Q$ such that (X_q, Y) is an extended Hilbert basis generating $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$ and Y is an Hilbert basis generating $C \cap \mathbb{Z}^n$. Once the sets X_q are computed, the constant and the periods of an extended Hilbert basis generating $P \cap \mathbb{Z}^n$ are computed thanks to Theorems 242 and 243.

We have the following theorem.

Theorem 246. *Let* $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\texttt{I}}, Q_{\texttt{F}})$ *be the reduced minimal NDD representing the set* $P \cap \mathbb{Z}^n$ *for some polyhedron* P *such that* $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\texttt{sign}}\}$ *and let* $\mathbf{C}$ *be an integer matrix such that* $\text{char-cone}(P) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{0}\}$*. With* $\mathcal{A}$ *and* $\mathbf{C}$ *as input parameters, the algorithm* COMPUTEBASIS *presented in Figure 7.10 terminates, and if* (X, Y) = COMPUTEBASIS*(*$\mathcal{A}$*,* $\mathbf{C}$*), then* (X, Y) *is an extended Hilbert basis generating* $P \cap \mathbb{Z}^n$*.*

The time complexity of the algorithm is $\mathcal{O}(2^{|\mathcal{A}|})$*.*

Proof. Let $C = \text{char-cone}(P)$. The sets X_q are computed incrementally. By construction, one explores only acyclic paths rooted at $q_{\texttt{I}}$, and for each $\mathbf{x} \in X_q$, there exists an encoding u of $\mathbf{x}$ labeling an acyclic path from $q_{\texttt{I}}$ to q.

At the lth iteration of the main **while** -loop at lines 11-23, one explores acyclic paths rooted at $q_{\texttt{I}}$ of length $l + 1$, more precisely, one considers all acyclic paths labeled by $w\alpha$ such that $|w| = l$ if $q = \hat{\delta}(q_{\texttt{I}}, w)$, then $\langle w \rangle_{r,n} \in X_q$ (practically, it suffices to store the minimal encodings, and their lengths, of all vectors in the sets X_q, $q \in Q$). If $\delta(q, \alpha) = q'$, one checks first whether $w\alpha$ labels an acyclic path in $\mathcal{A}$ rooted at $q_{\texttt{I}}$ via the function ACYCLIC? (it suffices to check whether a state is met twice in the path labeled by $w\alpha$). Then one checks whether there exists a

function COMPUTEBASIS(NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$, $\mathbf{C}$) : (**set of** integer vector, **set of** integer vector)

```
 1:  var W, W_next, Q_C : set of state;
 2:      q, q' : state;
 3:      B_per, B_cst, X_1, ..., X_|Q| : set of integer vector;
 4:      α, α_sign : symbol;
 5:      w : word;
 6:      x : integer vector;
 7:      l : integer;
 8:  begin
 9:      let q and α_sign such that δ(q_I, α_sign) = q;
10:      X_q := {⟨α_sign⟩_{r,n}}; W_next := {q}; l := 0;
11:      while W_next ≠ ∅ do
12:          begin
13:              W := W_next;
14:              W_next := ∅; l := l + 1;
15:              for each q ∈ W do
16:                  begin
17:                      for each q' ∈ Q, w ∈ (Σ_r^n)*, α ∈ Σ_r^n such that
                             δ(q, α) = q' ∧ ACYCLIC?(A, wα) ∧ |w| = l
                         ∧⟨w⟩_{r,n} ∈ X_q ∧ ∀y ∈ X_{q'} C⟨wα⟩_{r,n} − y ≰ 0 do
18:                          begin
19:                              X_{q'} := X_{q'} ∪ {⟨wα⟩_{r,n}};
20:                              W_next := W_next ∪ {q'};
21:                          end
22:                  end
23:          end

     (...)
```

Figure 7.10: Function COMPUTEBASIS

(...)

for each $q \in Q_{\mathtt{F}}$ **do** $B_{\mathtt{cst}} := B_{\mathtt{cst}} \cup X_q$;

let $Q_C := \{q \mid \hat{\delta}(q, \alpha_{\mathtt{sign}}^k) = q_{\mathtt{s}}$ for some sign-state $q_{\mathtt{s}}$ and $k \in \mathbb{N}\}$;

for each $q \in Q_C$ **do** $B_{\mathtt{per}} := B_{\mathtt{per}} \cup X_q)$;

let q **such that** $\delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = q$;

for each $q' \in Q, \alpha \in \Sigma_r^n, \mathbf{x} \in X_{q'}$ **such that** $\delta(q', \alpha) = q$ **do**

$B_{\mathtt{per}} := B_{\mathtt{per}} \cup \{r \cdot \mathbf{x} + \langle o\alpha \rangle_{r,n}\}$;

return ($B_{\mathtt{per}}$, $B_{\mathtt{cst}}$)

end

Figure 7.11: Function COMPUTEBASIS (continued)

vector $\mathbf{y} \in X_{q'}$ such that $\langle w\alpha \rangle_{r,n} - \mathbf{y} \in (C \cap \mathbb{Z}^n) \setminus \{\mathbf{0}\}$. If this is not the case, then $\langle w\alpha \rangle_{r,n}$ is added to $X_{q'}$ and q' is stored in $W_{\mathtt{next}}$ so that paths of length $l+1$ prefixed by $w\alpha$ are explored at the next iteration of the main **while** -loop.

Since one explores only acyclic paths of increasing length, there are at most $|Q|$ iterations of the main **while** -loop. By construction, when one leaves the main **while** -loop, $X_q \subseteq S_{\mathcal{A}}^q$. We show that for all states $q \in Q$, if $(X_{q,\min}, Y_{\min})$ is the minimal extended Hilbert basis generating $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$, then $X_{q,\min} \subseteq X_q$. The proof is by induction on the length of the minimal encoding w of the vectors $\mathbf{x} \in X_{q,\min}$, $q \in Q$.

- If $|w| = 1$, then, thanks to Lemma 192, $w = \alpha_{\mathtt{sign}}$, and by construction, $\langle w \rangle_{r,n}$ is added to X_q.

- If $|w| > 1$, then there exist an encoding w' and a symbol α such that $w'\alpha = w$. Since w is the minimal encoding of $\mathbf{x}$, w' is the minimal encoding of a vector $\mathbf{x}' \neq \mathbf{x}$. Let $q' \in Q$ such that $\hat{\delta}(q_{\mathtt{I}}, w') = q'$. Since $\mathbf{x} \in X_{q,\min}$, thanks to Lemma 244, $\mathbf{x}' \in X_{q',\min}$, and so, by hypothesis, $\mathbf{x}' \in X_{q'}$. By construction, there are at least $|w|$ iterations of the main **while** -loop, and at the $|w|$ iteration, the path labeled by w from $q_{\mathtt{I}}$ to q must has been considered. Since $\langle w \rangle_{r,n} \in X_{q,\min}$ and $X_q \subseteq S_{\mathcal{A}}^q$, by definition, for all $\mathbf{y} \in X_q$, $\langle w \rangle_{r,n} - \mathbf{y} \notin (C \cap \mathbb{Z}^n)$. So, $\mathbf{C}(\langle w \rangle_{r,n} - \mathbf{y}) \not\leq \mathbf{0}$. By inspection, we deduce that $\langle w \rangle_{r,n}$ is added to X_q.

The overall time complexity of the algorithm is $\mathcal{O}(2^{|\mathcal{A}|})$ since in the worst

case, one considers all acyclic paths rooted at $q_{\mathtt{I}}$. □

7.5 General Algorithm and Complexity

In this section, we summarize our main algorithm that, given a minimal reduced NDD $\mathcal{A}$ accepting the encodings of the integer elements of a polyhedron P, synthesizes a quantifier-free Presburger formula $\varphi(\mathbf{x})$ such that $\varphi(\mathbf{x})$ holds iff there is a word u with $\langle u \rangle_{r,n} = \mathbf{x}$ such that $u \in L(\mathcal{A})$.

The general idea is to decompose the polyhedron P into pointed polyhedra $P_{\alpha_{\mathtt{sign}}}$ according to the sign of the vector components. More precisely, for each sign symbol $\alpha_{\mathtt{sign}}$, we compute the reduced minimal NDD $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ such that $L(\mathcal{A}_{\alpha_{\mathtt{sign}}}) = \{\alpha_{\mathtt{sign}} u \in L(\mathcal{A})\}$, and $P_{\alpha_{\mathtt{sign}}}$ is the convex hull of the vectors whose encodings are accepted by $\mathcal{A}_{\alpha_{\mathtt{sign}}}$. Then for each $\mathcal{A}_{\alpha_{\mathtt{sign}}}$, the algorithm proceeds in two steps, the generation of a matrix $\mathbf{C}_{\alpha_{\mathtt{sign}}}$ such that

$$\mathbf{x} \in \text{char-cone}(P_{\alpha_{\mathtt{sign}}}) \Leftrightarrow \mathbf{C}_{\alpha_{\mathtt{sign}}} \mathbf{x} \leq \mathbf{0},$$

and the computation of an extended Hilbert basis $(X_{\alpha_{\mathtt{sign}}}, Y_{\alpha_{\mathtt{sign}}})$ generating $P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n$. A quantifier-free Presburger formula corresponding to $P \cap \mathbb{Z}^n$ is then $\varphi(\mathbf{x})$ with

$$\varphi(\mathbf{x}) =_{def} \bigvee_{\alpha_{\mathtt{sign}} \in \{0, r-1\}^n} \left(\bigvee_{\mathbf{x}' \in X_{\alpha_{\mathtt{sign}}}} \left(\mathbf{C}_{\alpha_{\mathtt{sign}}} (\mathbf{x} - \mathbf{x}') \leq \mathbf{0} \right) \right).$$

The formal algorithm is given in Figure 7.12.

In order to prove the correctness of the algorithm given in Figure 7.12, we need the following lemma which shows how to decompose the NDD representing the integer elements of a polyhedron P into NDDs regrouping the integer elements of P sharing the same sign.

Lemma 247. *Let $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$ be the reduced minimal NDD representing the set $P \cap \mathbb{Z}^n$ for some polyhedron P, and let $\alpha_{\mathtt{sign}} \in \{0, r-1\}^n$ be a sign symbol.*

Let $\mathcal{A}_{\alpha_{\mathtt{sign}}} = (Q \cup \{q'_{\mathtt{I}}\}, \Sigma_r^n, \delta_{\alpha_{\mathtt{sign}}}, , Q_{\mathtt{F}})$ with $\delta_{\alpha_{\mathtt{sign}}}(q, \alpha) = \delta(q, \alpha)$ for all $q \in Q$ and $\alpha \in \Sigma_r^n$, and $\delta_{\alpha_{\mathtt{sign}}}(q'_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = \delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}})$.

The automaton $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is a deterministic NDD in strong normal form representing the set $P \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$.

function GENERATEFORMULA(NDD $\mathcal{A} = (Q, \Sigma_r^n, \delta, q_{\mathtt{I}}, Q_{\mathtt{F}})$) : Presburger formula

1: **var** $\mathcal{A}_{\alpha_{\mathtt{sign}}}$: automaton;
2: $\delta_{\alpha_{\mathtt{sign}}}$: function;
3: $\mathbf{C}_{\alpha_{\mathtt{sign}}}$: matrix;
4: φ : Presburger formula;
5: $\alpha, \alpha_{\mathtt{sign}}$: symbol;
6: $X_{\alpha_{\mathtt{sign}}}, Y_{\alpha_{\mathtt{sign}}}$: **set of** integer vector;
7: **begin**
8: $\varphi(\mathbf{x})$:= **false**;
9: **for each** $\alpha_{\mathtt{sign}} \in \{0, r-1\}^n$ **do**
10: **begin**
11: **let** $q \notin Q$;
12: **let** $\delta_{\alpha_{\mathtt{sign}}} : (Q \cup \{q\}) \times \Sigma_r^n \to Q \cup \{q\}$
$$: \begin{cases} (q' \neq q, \alpha) \to \delta(q, \alpha) \\ (q, \alpha_{\mathtt{sign}}) \to \delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) \end{cases};$$
13: $\mathcal{A}_{\alpha_{\mathtt{sign}}}$:= AUTO_MINIMIZE($Q \cup \{q\}, \Sigma_r^n, \delta_{\alpha_{\mathtt{sign}}}, q, Q_{\mathtt{F}}$);
14: **if** AUTO_EMPTY?($\mathcal{A}$) = **false then**
15: **begin**
16: **let** $\mathbf{C}_{\alpha_{\mathtt{sign}}}$ **such that**
$\mathbf{C}_{\alpha_{\mathtt{sign}}}\mathbf{x} \leq \mathbf{0}$ = CHARCONEFORMULA($\mathbf{A}_{\alpha_{\mathtt{sign}}}$);
17: $(X_{\alpha_{\mathtt{sign}}}, Y_{asign})$:= COMPUTEBASIS($\mathcal{A}_{\alpha_{\mathtt{sign}}}$, $\mathbf{C}_{\alpha_{\mathtt{sign}}}$);
18: $\varphi(\mathbf{x}) := \varphi(\mathbf{x}) \vee \bigvee_{\mathbf{x}' \in X_{\alpha_{\mathtt{sign}}}} \mathbf{C}_{\alpha_{\mathtt{sign}}}(\mathbf{x} - \mathbf{x}') \leq \mathbf{0}$;
19: **end**
20: **end**
21: **return** φ
22: **end**

Figure 7.12: Function GENERATEFORMULA

Proof. Clearly, $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is deterministic and in strong normal form, and $L(\mathcal{A}_{\alpha_{\mathtt{sign}}}) \subseteq \{\alpha_{\mathtt{sign}} u \mid u \in (\Sigma_r^n)^*\}$.

- If for all $\alpha u \in L(\mathcal{A})$, $\alpha \neq \alpha_{\mathtt{sign}}$, then there is no outgoing transition from q and since $\mathcal{A}$ is deterministic in strong normal form, $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is also deterministic in strong normal form.

- Suppose that $\alpha_{\mathtt{sign}} u \in L(\mathcal{A})$ for some $u \in (\Sigma_r^n)^*$. Let $q_{\alpha_{\mathtt{sign}}} \in Q$ such that $\delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = q_{\alpha_{\mathtt{sign}}}$. By definition, $\delta(q'_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = q_{\alpha_{\mathtt{sign}}}$. Also, by definition, $\hat{\delta}(q_{\alpha_{\mathtt{sign}}}, u) \in Q_{\mathtt{F}}$, and therefore, $\hat{\delta}_{\alpha_{\mathtt{sign}}}(q_{\alpha_{\mathtt{sign}}}, u) \in Q_{\mathtt{F}}$. So, $\alpha_{\mathtt{sign}} u \in L(\mathcal{A}_{\alpha_{\mathtt{sign}}})$, i.e.

$$L(\mathcal{A}_{\alpha_{\mathtt{sign}}}) \supseteq L(\mathcal{A}) \cap \{\alpha_{\mathtt{sign}} u \mid u \in (\Sigma_r^n)^*\}. \tag{7.54}$$

  Conversely, if $\alpha v \in L(\mathcal{A}_{\alpha_{\mathtt{sign}}})$, then $\hat{\delta}_{\alpha_{\mathtt{sign}}}(q_{\alpha_{\mathtt{sign}}}, v) \in Q_{\mathtt{F}}$. So, by definition, $\hat{\delta}(q_{\alpha_{\mathtt{sign}}}, v) \in Q_{\mathtt{F}}$ and $\alpha v \in L(\mathcal{A})$, i.e.

$$L(\mathcal{A}_{\alpha_{\mathtt{sign}}}) \subseteq L(\mathcal{A}) \cap \{\alpha_{\mathtt{sign}} u \mid u \in (\Sigma_r^n)^*\}. \tag{7.55}$$

We conclude that

$$L(\mathcal{A}_{\alpha_{\mathtt{sign}}}) = L(\mathcal{A}) \cap \{\alpha_{\mathtt{sign}} u \mid u \in (\Sigma_r^n)^*\}, \tag{7.56}$$

and by definition, $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is an NDD. □

Theorem 248. *Given a reduced minimal NDD $\mathcal{A}$ representing the integer elements of a polyhedron P, the algorithm* GENERATEFORMULA *presented in Fig 7.12 terminates, and if $\varphi(\mathbf{x})$ is the Presburger formula returned by* GENERATEFORMULA*($\mathcal{A}$), then we have*

$$\varphi(\mathbf{x}) \ \textit{iff}\ \mathbf{x} \in P \cap \mathbb{Z}^n.$$

The time complexity of the algorithm is $\mathcal{O}(2^n \cdot 2^{|\mathcal{A}|})$.

Proof. Thanks to Lemma 247, for each $\alpha_{\mathtt{sign}} \in \{0, r-1\}^n$, $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is the reduced minimal NDD representing the set $S_{\alpha_{\mathtt{sign}}} = \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{x} \in P \wedge \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$. By construction, the NDD $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ has at most one state more than $\mathcal{A}$.

Let $P_{\alpha_{\mathtt{sign}}}$ be the convex hull of $S_{\alpha_{\mathtt{sign}}}$. By definition, $P_{\alpha_{\mathtt{sign}}}$ is a polyhedron and $P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n = S_{\alpha_{\mathtt{sign}}}$.

With $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ as input, the function CHARCONEFORMULA, described in Fig.7.4, generates a system of linear inequations $\mathbf{C}_{\alpha_{\mathtt{sign}}} \mathbf{x} \leq \mathbf{0}$ in time polynomial in

$|\mathcal{A}_{\alpha_{\mathtt{sign}}}|$, such that for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{x} \in \text{char-cone}(P_{\alpha_{\mathtt{sign}}})$ if and only if $\mathbf{C}_{\alpha_{\mathtt{sign}}}\mathbf{x} \leq \mathbf{0}$, as proved in Theorem 190. Thanks to Theorem 246, $(X_{\alpha_{\mathtt{sign}}}, Y_{\alpha_{\mathtt{sign}}})$ is an extended Hilbert basis generating $P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n$, and the computation is done in time proportional to $2^{|\mathcal{A}_{\alpha_{\mathtt{sign}}}|}$ in the worst case.

Finally, thanks to Lemma 38 and Theorem 41, $\text{char-cone}(P_{\alpha_{\mathtt{sign}}}) \cap \mathbb{Z}^n = \text{cone}_{\mathbb{Z}}(Y_{\alpha_{\mathtt{sign}}})$ and by hypothesis, $\text{char-cone}(P_{\alpha_{\mathtt{sign}}}) \cap \mathbb{Z}^n = \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{C}_{\alpha_{\mathtt{sign}}}\mathbf{x} \leq \mathbf{0}\}$. Also, by definition, $P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n = \bigcup_{\mathbf{x}' \in X_{\alpha_{\mathtt{sign}}}} \mathbf{x}' + \text{cone}_{\mathbb{Z}}(Y)$. So, we have

$$P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n = \left\{ \mathbf{x} \in \mathbb{Z}^n \mid \bigvee_{\mathbf{x}' \in X_{\alpha_{\mathtt{sign}}}} \left(\mathbf{C}_{\alpha_{\mathtt{sign}}}(\mathbf{x} - \mathbf{x}') \leq \mathbf{0} \right) \right\}.$$

Since $P \cap \mathbb{Z}^n = \bigcup_{\alpha_{\mathtt{sign}} \in \{0,r-1\}^n} \left(P_{\alpha_{\mathtt{sign}}} \cap \mathbb{Z}^n \right)$, by inspection, it is immediate that for all $\mathbf{x} \in \mathbb{Z}^n$, $\varphi(\mathbf{x})$ holds iff $\mathbf{x} \in P$. □

The exponential time complexity of the algorithm generating the basis is related to the fact that in the worst case, one explores all acyclic paths rooted in the initial state of the NDD.

In practice, a prototype of our algorithm performs well on large automata, as shown in Section 7.6. We attribute this to the fact that our algorithm succeeds in filtering out many irrelevant paths. Based on the description of our algorithm, if the sizes of the sets X_q, $q \in Q$, are proportional to the size of the minimal extended Hilbert basis, both in terms of number of elements and of minimal length of words encoding the largest element in the set, the actual cost would be polynomial in the size of the NDD, in the number of elements in the basis and in the minimal encoding lengths of the elements, and this is what suggest the experimental results presented in the following section.

Finally, in the particular case where $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$ and $P \cap \mathbb{Z}^n = \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (\text{char-cone}(P) \cap \mathbb{Z}^n)$, a formula for $P \cap \mathbb{Z}^n$ is given by $\varphi(\mathbf{x}) \equiv \mathbf{C}(\mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) \leq \mathbf{0}$, and therefore, in this case, the formula for $P \cap \mathbb{Z}^n$ is generated in time polynomial in $|\mathcal{A}|$ since it is not required to compute the basis of $P \cap \mathbb{N}^n$, it suffices to apply the function CHARCONEFORMULA.

Note that testing whether $P \cap \mathbb{Z}^n \subseteq \{\mathbf{x} \in \mathbb{Z}^n \mid \text{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$ and $P \cap \mathbb{Z}^n = \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (\text{char-cone}(P) \cap \mathbb{Z}^n)$, can be done in time linear in $|\mathcal{A}|$ by checking first if there is only one transition outgoing from $q_{\mathtt{I}}$ and that this transition is labeled by $\alpha_{\mathtt{sign}}$, and secondly, whether $Q_C = Q_{\mathtt{F}}$ where Q_C is the set $Q_C = \{q \in Q \mid \exists p \in \mathbb{N}\ \exists q_{\mathtt{s}} \in Q(\hat{\delta}(q, \alpha_{\mathtt{sign}}^p) = q_{\mathtt{s}} \wedge q_{\mathtt{s}} \text{ is a sign-state})\}$, as deduced from Theorem 199.

7.6 Experimental Results

The algorithms presented in this chapter have been implemented within the LASH library [LAS]. Note that the algorithms have been slightly modified in order to use the synchronous interleaved encoding scheme $E_{I(r)}$, which significantly decreases the running time. Also, the tests have been performed on sets of positive integer vectors, and therefore, only the sign symbol o needs to be considered.

The following sets have been converted into NDDs over which we have run our implementation. As encoding basis, we have taken $r = 2$. Note that the sets S_1,S_2, S_3,S_4,S_5,S_6 have been taken from [AC97], (S_6 is an example which is not handled efficiently in [AC97] because their pruning criterion $\mathcal{C}1$ does not apply).

$$
\begin{aligned}
S_1 &= \{\mathbf{x} \in \mathbb{Z}^4 \mid \begin{bmatrix} 1 & -1 & -1 & -3 \\ -2 & 3 & 3 & -5 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 2 \\ 3 \end{bmatrix}\} \\
S_2 &= \{\mathbf{x} \in \mathbb{Z}^5 \mid \begin{bmatrix} 7 & -2 & 11 & 3 & -5 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 5 \end{bmatrix}\} \\
S_3 &= \{\mathbf{x} \in \mathbb{Z}^4 \mid \begin{bmatrix} 1 & -2 & -3 & 4 \\ 100 & 45 & -78 & -67 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 0 \\ 0 \end{bmatrix}\} \\
S_4 &= \{\mathbf{x} \in \mathbb{Z}^5 \mid \begin{bmatrix} 23 & -56 & -34 & 12 & 11 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 0 \end{bmatrix}\} \\
S_5 &= \{\mathbf{x} \in \mathbb{Z}^4 \mid \begin{bmatrix} 1 & 0 & -4 & 8 \\ -1 & 0 & 4 & -8 \\ 12 & 19 & -11 & -7 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 2 \\ -2 \\ -7 \end{bmatrix}\} \\
S_6 &= \{\mathbf{x} \in \mathbb{Z}^3 \mid =_{def} \begin{bmatrix} 23 & -12 & -9 \\ 1 & -8 & -8 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 0 \\ 0 \end{bmatrix}\} \\
S_7 &= \{\mathbf{x} \in \mathbb{Z}^7 \mid \begin{bmatrix} 3 & -7 & -1 & -1 & -2 & 0 & -1 \\ 4 & -3 & 9 & 3 & -5 & -3 & 1 \\ 5 & 1 & 0 & 0 & 0 & 0 & -4 \\ -5 & -1 & 0 & 0 & 0 & 0 & 4 \\ 0 & 1 & 3 & 0 & 0 & -1 & 0 \\ 0 & -1 & -3 & 0 & 0 & 1 & 0 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 10 \\ 5 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}\} \\
S_8 &= \{\mathbf{x} \in \mathbb{Z}^5 \mid \begin{bmatrix} 1 & 1 & 1 & -1 & -4 \\ 4 & -3 & 9 & 3 & -5 \\ 1 & -1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 10 \\ 5 \\ 3 \\ -3 \\ -10 \\ -5 \end{bmatrix}\} \\
S_9 &= \{\mathbf{x} \in \mathbb{Z}^7 \mid \begin{bmatrix} 1 & -4 & 1 & 0 & 0 & -4 & -1 \\ 0 & 0 & 0 & 3 & -1 & 0 & 3 \\ 0 & 1 & -2 & 6 & 0 & 0 & 0 \\ 0 & -1 & 2 & -6 & 0 & 0 & 0 \\ 13 & -1 & 0 & -11 & 1 & 0 & 0 \\ -13 & 1 & 0 & 11 & -1 & 0 & 0 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}\}
\end{aligned}
$$

$$S_{10} = \{\mathbf{x} \in \mathbb{Z}^7 \mid \begin{bmatrix} 1 & -4 & 1 & 0 & 0 & -4 & -1 \\ 0 & 0 & 0 & 3 & -1 & 0 & 3 \\ 0 & 1 & -2 & 6 & 0 & 0 & 0 \\ 0 & -1 & 2 & -6 & 0 & 0 & 0 \\ 13 & -1 & 0 & -11 & 1 & 0 & 0 \\ -13 & 1 & 0 & 11 & -1 & 0 & 0 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 10 \\ 17 \\ 6 \\ -6 \\ 1 \\ -1 \end{bmatrix} \}$$

$$S_{11} = \{\mathbf{x} \in \mathbb{Z}^6 \mid \begin{bmatrix} 1 & 1 & -4 & 5 & 0 & 0 \\ 0 & 0 & 0 & 3 & -1 & 3 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 11 & -8 & 0 & -1 & 0 & 3 \\ -1 & 0 & -2 & -1 & 0 & 0 \\ 0 & 0 & 1 & 6 & -5 & 0 \\ 0 & 0 & -1 & -6 & 5 & 0 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 10 \\ -10 \\ 4 \\ -4 \\ 0 \\ -5 \\ 3 \\ -3 \end{bmatrix} \}$$

$$S_{12} = \{\mathbf{x} \in \mathbb{Z}^6 \mid \begin{bmatrix} 7 & 2 & -3 & -5 & 0 & -1 \\ 0 & 0 & 0 & 3 & -1 & -6 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & -1 & 0 & 1 \\ -4 & 0 & 1 & 0 & 0 & 3 \\ 5 & 0 & 1 & 0 & 0 & 6 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 10 \\ -10 \\ 3 \\ -3 \\ 0 \\ 0 \\ 12 \end{bmatrix} \}$$

$$S_{13} = \{\mathbf{x} \in \mathbb{Z}^3 \mid \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \mathbf{x} \leq \begin{bmatrix} 31 \\ 31 \\ 31 \end{bmatrix} \}$$

The following table shows the results of our experiments. The columns give successively the name of the set, the number of states of the corresponding NDD, the number of constants and periods in the basis as well as the maximal length of minimal encodings of elements in the basis, the time (in seconds) and memory (in megabytes) required for the generation of a formula corresponding to the characteristic cone, and the time and the memory required for the computation of the basis.

	$\|Q\|$	Extended Hilbert Basis			CHARCONEFORMULA		COMPUTEBASIS	
		Constant	Periods	length	time	space	time	space
					[Secs.]	[Mbytes]	[Secs]	[Mbytes]
S_1	425	69	214	4	0.1	0	0.1	0
S_2	198	29	39	4	0.1	0	0.0	0
S_3	11609	1	26	10	2.1	5	0.1	3
S_4	889	1	567	6	0.1	0	0.3	1
S_5	3439	8	21	7	0.2	1	0.2	1
S_6	165	1	15	5	0.0	0	0.0	0
S_7	8465	25	37	6	0.1	8	0.3	5
S_8	2368	95	26	5	0.2	8	0.1	1
S_9	99094	1	246	9	10.5	32	7.0	86
S_{10}	132619	1040	246	10	12.5	34	10.5	112
S_{11}	43777	25541	922	10	3.7	10	537.8	116
S_{12}	7496	88	19	5	0.8	2	0.1	2
S_{13}	21	32768	0	5	0.1	0	19.5	5

From these results, we can see that the algorithm performs well on most sets, especially the generation of the formula corresponding to the characteristic cone, which is done for each set in a few seconds. This fits well with the computed complexity of this part of the algorithm. Indeed, in Section 7.2.4, we showed that if the number of zero-SCCs is small compared to the number of states, which is the case in practice, the computation is proportional to the number of transitions. Regarding the generation of the basis, our algorithm performs well on most sets. The time cost appears proportional to the maximal length of minimal encodings of elements in the basis, the number of elements in the basis, and the size of the NDD (when using the serial encoding in basis r, we have $|\mathcal{A}| = r \cdot |Q|$). This suggests that in the algorithm of COMPUTEBASIS of Fig. 7.10, for each $q \in Q$, the size of X_q is proportional to the size of the minimal extended Hilbert basis.

7.7 Conclusion

In this chapter, we first have characterized finely the structure of an NDD $\mathcal{A}$ representing the integer elements of a polyhedron $P = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{b}\}$ such that the encodings of all integer elements have the same sign symbol.

Then, we have presented an algorithm, CHARCONEFORMULA, relying heavily on the structural properties of $\mathcal{A}$, which synthesizes a linear system of inequations $\mathbf{C}\mathbf{x} \leq \mathbf{0}$ corresponding to the characteristic cone of P, in linear time when assuming that the number of strongly connected components with a sign-state is

small compared to the number of states, which is verified in practice.

We have also presented an algorithm, COMPUTEBASIS, extracting from $\mathcal{A}$ an extended Hilbert basis (X, Y) generating $P \cap \mathbb{Z}^n$. In the worst case, the computational costs of the algorithm are proportional to the number of acyclic paths in $\mathcal{A}$ rooted at the initial state. In practice, our method filters out many acyclic paths. A quantifier-free Presburger formula is directly obtained from X and from the linear system $\mathbf{C}\mathbf{x} \leq \mathbf{0}$, and has the form

$$\bigcup_{\mathbf{x}_i \in X} \mathbf{C}(\mathbf{x} - \mathbf{x}_i) \leq \mathbf{0}.$$

Then, we presented an algorithm, GENERATEFORMULA, which is not restricted to NDDs accepting sets whose elements have all the same sign, and this is done as follows. Given an NDD $\mathcal{A}$ representing the integer element of a polyhedron P, we consider each sign symbol $\alpha_{\mathtt{sign}}$ in turn. In each case, we compute in linear time a deterministic NDD $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ accepting the integer elements of P whose encodings have $\alpha_{\mathtt{sign}}$ as sign symbol. This set is the integer elements of a polyhedron $P_{\alpha_{\mathtt{sign}}}$ such that all elements in $P \cap \mathbb{Z}^n$ have the same sign. Once $\mathcal{A}_{\alpha_{\mathtt{sign}}}$ is minimized, we apply the algorithms CHARCONEFORMULA and GENERATEBASIS and compute respectively a system of linear equations $\mathbf{C}_{\alpha_{\mathtt{sign}}}\mathbf{x} \leq \mathbf{0}$ corresponding to the characteristic cone of $P_{\alpha_{\mathtt{sign}}}$ and an extended Hilbert basis $(X_{\alpha_{\mathtt{sign}}}, Y_{\alpha_{\mathtt{sign}}})$ of $P \cap \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$.

A quantifier-free Presburger formula corresponding to $P \cap \mathbb{Z}^n$ is then $\varphi(\mathbf{x})$ with

$$\varphi(\mathbf{x}) =_{def} \bigvee_{\alpha_{\mathtt{sign}} \in \{0, r-1\}^n} \left(\bigvee_{\mathbf{x}' \in X_{\alpha_{\mathtt{sign}}}} \left(\mathbf{C}_{\alpha_{\mathtt{sign}}}(\mathbf{x} - \mathbf{x}') \leq \mathbf{0} \right) \right).$$

The overall algorithm has been tested with a prototype implementation, and the experimental results are very encouraging : the generation of formulas and bases corresponding to NDDs with more than 100,000 states can be achieved in seconds. Experimental results suggest that the actual cost is proportional to the size of the NDD as well as to both the number of elements in the basis and their encoding lengths.

7.7.1 Related Work

In [Lug04], an algorithm is proposed which, given a deterministic NDD $\mathcal{A}$ representing a set $S_{\mathcal{A}} \subseteq \mathbb{N}^n$, computes a pair of finite sets $(S_{\mathtt{cst}}, S_{\mathtt{per}})$ such that $\mathbf{x} \in S_{\mathcal{A}}$ iff $\mathbf{x} \in \bigcup_{\mathbf{x}_c \in S_{\mathtt{cst}}} \mathbf{x}_c + \mathrm{cone}_{\mathbb{Z}}(S_{\mathtt{per}})$. The restrictions on the input NDD is that the

set $S_{\mathcal{A}}$ must correspond to a set of the form $\bigcup_{\mathbf{x}_c \in S_{\mathtt{cst}}} \mathbf{x}_c + \mathrm{cone}_{\mathbb{Z}}(S_{\mathtt{per}})$ for some finite sets $S_{\mathtt{cst}}$ and $S_{\mathtt{per}}$, and that $\mathrm{cone}_{\mathbb{Q}}(S_{\mathtt{per}})$ must be pointed. Clearly, any pair $(S_{\mathtt{cst}}, S_{\mathtt{per}})$ is not necessarily an extended Hilbert basis since we do not require $S_{\mathtt{per}}$ to be an Hilbert basis.

In a way similar to what we have done regarding extended Hilbert basis, there is a minimality criterion for the pairs $(S_{\mathtt{cst}}, S_{\mathtt{per}})$ since the cone $\mathrm{cone}_{\mathbb{Q}}(S_{\mathtt{per}})$ is pointed, and there exist formulas $\varphi_{\mathtt{cst}}$ and $\varphi_{\mathtt{per}}$ expressed in Presburger arithmetic extended with a predicate corresponding to the membership to $S_{\mathcal{A}}$ such that for all $\mathbf{x} \in \mathbb{Z}^n$, $\varphi_{\mathtt{cst}}(\mathbf{x})$ holds iff $\mathbf{x} \in S_{\mathtt{cst}}$ and similarly $\varphi_{\mathtt{per}}(\mathbf{x})$ holds iff $\mathbf{x} \in S_{\mathtt{per}}$. Interestingly, the algorithm does not depend on the implementation details associated to the NDDs, such as the encoding scheme. The shortcoming of the method is its huge computational cost, $\mathcal{O}(2^{2^{|\mathcal{A}|}})$ which in practice prohibits any practical application, as seen in Section 7.1.

More recently, [Ler04b, Ler05] detail a polynomial algorithm computing a Presburger formula corresponding to the set represented by a deterministic NDD using the reverse synchronous encoding scheme. The sole restriction on the input NDD is that all elements in the represented set must be positive integer vectors, i.e. vectors in $\mathbb{N}^n$ for some $n \in \mathbb{N}$. Although the algorithm is polynomial, its costs appears to be high in practice. Indeed, the computational costs are proportional to at least $|Q|^4$, where Q is the set of states, and this leads to huge numbers when considering NDDs with thousands of states. Another aspect which may prevent the practical application of the algorithm is the presence of a polynomial number of quantifiers in the generated formula.

7.8 Additional Proof Details

7.8.1 Proof of Theorem 177

In order to prove Theorem 177, we need some additional lemmas.

First, we recall some definitions and theorems regarding congruences.

Let $\mathbf{a}, \mathbf{x} \in \mathbb{Z}^n, b \in \mathbb{Z}, m \in \mathbb{N}$ with $m \neq 0$. A *congruence* $\mathbf{a}.\mathbf{x} \equiv_m b$ expresses that $\mathbf{a}.\mathbf{x} - b$ is divisible by m. The number m is called the *modulus*. A system of linear congruences is a conjunction of congruences $\mathbf{a}_i.\mathbf{x} \equiv_m b_i$ and is represented in matrix form as $\mathbf{A}\mathbf{x} \equiv_m \mathbf{b}$ for some integer matrix $\mathbf{A}$ and integer vector $\mathbf{b}$.

A system of p linear congruences with n indeterminates is *redundant* if $p > n$ and *defective* if $p < n$. Two solutions $\mathbf{x}_1, \mathbf{x}_2$ of a system of linear congruences

$\mathbf{Ax} \equiv_m \mathbf{b}$ are *congruous* if for all i, $\mathbf{x}_1[i] \equiv_m \mathbf{x}_2[i]$, otherwise, they are *incongruous*.

Theorem 249. *If every determinant of the augmented matrix of a redundant system of linear congruences $\mathbf{Ax} \equiv_m \mathbf{b}$ is divisible by the modulus q, while the greatest divisor of the unaugmented matrix is prime to the modulus, the system is resoluble and admits only one incongruous solution.*

Proof. See [Smi61]. □

Theorem 250. *If the greatest common divisor of the determinants of the unaugmented matrix of a non-redundant, non-defective system of linear congruences $\mathbf{Ax} \equiv_m \mathbf{b}$ is not divisible by the modulus q, the system is resoluble and admits only one incongruous solution.*

Proof. See [Smi61]. □

Lemma 251. *Let $\mathbf{A} \in \mathbb{Z}^{n\times d}$ be a prime matrix with $n \geq d$. For all $0 < m \in \mathbb{N}$, and $\mathbf{x} \in \mathbb{Z}^d$, $\mathbf{Ax} \equiv_m \mathbf{0} \Leftrightarrow \mathbf{x} \equiv_m \mathbf{0}$.*

Proof. Suppose $\mathbf{Ax} \equiv_m \mathbf{0}$. Since A is prime, according to Theorems 249 and 250, there is one and only one incongruous solution to $\mathbf{Ax} \equiv_m \mathbf{0}$. Since $\mathbf{0}$ is a solution, for all solutions $\mathbf{x}$, $\mathbf{x} \equiv_m \mathbf{0}$.

Suppose now that $\mathbf{x} \equiv_m \mathbf{0}$. Then it is obvious that for any $\mathbf{a} \in \mathbb{Z}^d$, $\mathbf{a}.\mathbf{x} \equiv_m \mathbf{0}$, and therefore, $\mathbf{Ax} \equiv_m \mathbf{0}$. □

Next, we prove than for any linear system $\mathbf{Ax} = \mathbf{0}$, whenever there is a complete set of positive integer solutions, there is a fundamental set of positive integer solutions.

Lemma 252. *If $\{\mathbf{y}_1, \ldots, \mathbf{y}_d\}$ is a complete set of solutions of the linear system $\mathbf{Ax} = \mathbf{0}$ and $\mathbf{y}_i \in \mathbb{N}^n$ for $i \in \{1, \ldots, d\}$, then there exists a fundamental set of solutions $\mathbf{y}_1^f, \ldots, \mathbf{y}_d^f$ such that $\mathbf{y}_i^f \in \mathbb{N}^n$ for $i \in \{1, \ldots, d\}$.*

Proof. We may suppose that the components of the vector $\mathbf{y}_1, \ldots, \mathbf{y}_d \in \mathbb{Z}^n$ admit no common divisor but unity; for if all components of a vector $\mathbf{y}_i$ are divisible by k_i, the vectors $\mathbf{y}_1/k_1, \ldots, \mathbf{y}_d/k_d$ are also a complete set of positive independent solutions for $\mathbf{Ax} = \mathbf{0}$.

We will construct a sequence of matrices $\mathbf{Y}^{(i)}$, $1 \leq i \leq d$, such that $\mathbf{Y}^{(1)} = (\mathbf{y}_1)$, and for each i, $\mathbf{Y}^{(i)} \in \mathbb{Z}^{n\times i}$ and is prime, and the columns of $\mathbf{Y}^{(i)}$ are a set of positive independent solutions for $\mathbf{Ax} = \mathbf{0}$. So, the columns of $\mathbf{Y}^{(d)}$ form a fundamental set of solutions for $\mathbf{Ax} = \mathbf{0}$.

The idea is to augment the matrix $\mathbf{Y}^{(k)}$, which is supposed to be prime, by a linear combination of the columns of $\mathbf{Y}^{(k)}$ and $\mathbf{y}_{k+1}$ such that the resulting matrix is prime. Let the determinants of the matrix $\mathbf{Y}^{(k)}$ augmented with $\mathbf{y}_{k+1}$ admit μ as greatest divisor. Determine $x_1, \ldots, x_k$ by the system of congruences

$$\mathbf{Y}^{(k)} \begin{bmatrix} x_1 \\ \vdots \\ x_k \end{bmatrix} \equiv_\mu \mathbf{y}_{k+1}$$

which is always soluble from Theorem 249 since $\mathbf{Y}^{(k)}$ is prime by recursive hypothesis.

We can always choose the $x_1, \ldots, x_k$ to be negative by subtracting a finite number of times μ.

Let $\mathbf{Y}^{(k+1)}$ be $\mathbf{Y}^{(k)}$ augmented with the vector $\mathbf{y}$ such that

$$\mathbf{y} = \frac{1}{\mu}\big(\mathbf{y}_{k+1} - \sum_{j=0}^{k-1} x_j \mathbf{Y}^{(k)}[*, j]\big)$$

By construction, $\mathbf{y} \in \mathbb{Z}^n$ and $\mathbf{Y}^{(k+1)}$ is prime. Indeed, adding to one column of a square matrix a linear combination of other columns does not modify the determinant, and the determinant of a square matrix is divided by μ if all elements of a column are divided by μ.

In addition, since the elements of $\mathbf{Y}^{(k)}$ and of $\mathbf{y}_{k+1}$ are nonnegative integers by hypothesis, the elements of $\mathbf{y}$ are nonnegative integers. Finally, $\mathbf{Y}^{(k+1)}$ is an independent set of solutions of $\mathbf{Ax} = \mathbf{0}$. Indeed, the k first columns are the columns of $\mathbf{Y}^{(k)}$ forming an independent set of solutions by recursive hypothesis, and $\mathbf{y}$ is a solution since it is a linear combination of solutions and it is independent of the first k columns since $\mathbf{y}_{k+1}$ is linearly independent of $\mathbf{y}_1 \ldots \mathbf{y}_k$. □

Lemma 253. *Let $\mathcal{S}$ be a zero-SCC and let $q_\mathbf{z}$ be the zero-state of $\mathcal{S}$ and let $d = \dim(\mathcal{S})$. If $0 < d < n$, there exists a prime matrix $\mathbf{Y} \in N^{n \times d}$ such that for all states $q \in \mathcal{S}$ and words $v_{q_\mathbf{z} q} \in L_\mathcal{A}(q_\mathbf{z} \to q)$,*

$$\mathrm{aff}_\mathbb{Q}(\{\langle ov \rangle_{r,n} \mid v \in L_\mathcal{A}(q_\mathbf{z} \to q)\}) \cap \mathbb{Z}^n = \{\langle ov_{q_\mathbf{z} q} \rangle_{r,n} + \mathbf{Yc} \mid \mathbf{c} \in \mathbb{Z}^d\}.$$

Proof. By definition, there exist $u_1, \ldots, u_d \in L_\mathcal{A}(q_\mathbf{z} \to q_\mathbf{z})$ such that the vectors $\langle ou_1 \rangle_{r,n}, \ldots, \langle ou_d \rangle_{r,n}$ are linearly independent.

From Corollary 174, there exists an integer matrix $\mathbf{B}$ such that $\mathrm{rank}(\mathbf{B}) = n - d$ and for all states $q \in \mathcal{S}$ and for all $v_{q_\mathbf{z} q} \in L_\mathcal{A}(q_\mathbf{z} \to q)$,

- $\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\} = \{\mathbf{x} \in \mathbb{N}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}$,
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\}$.

By definition, $\mathbf{B}\langle ou\rangle_{r,n} = \mathbf{0}$ for $u \in \{u_1, \ldots, u_d\}$, and since $\mathrm{rank}(\mathbf{B}) = n-d$ and $\langle ou_1\rangle_{r,n}, \ldots, \langle ou_d\rangle_{r,n}$ are linearly independent, the vectors $\langle ou_1\rangle_{r,n}, \ldots, \langle ou_d\rangle_{r,n}$ form a complete set of independent solutions of $\mathbf{Bx} = \mathbf{0}$.

From Lemma 252, there exists a fundamental set of natural solutions of $\mathbf{Bx} = \mathbf{0}$, and let $\{\mathbf{y}_1, \ldots, \mathbf{y}_d\}$ be such a set. Let $\mathbf{Y} \in \mathbb{N}^{n\times d}$ with $\mathbf{Y}[i,j] = \mathbf{y}_j[i]$, for $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, d\}$. By definition, $\mathbf{Y}$ is prime and $\{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{Bx} = \mathbf{0}\} = \{\mathbf{Yc} \mid \mathbf{c} \in \mathbb{Z}^d\}$.

Let q be a state of $\mathcal{S}$ and let $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. By definition of $\mathbf{B}$, we have

$$\begin{aligned}\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) \cap \mathbb{Z}^n &= \{\mathbf{x} \in \mathbb{Z}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\} \\ &= \{\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \sum_{i=1}^{d} \mu_i \mathbf{y}_i \mid \mu_i \in \mathbb{Z}\} \\ &= \{\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc} \mid \mathbf{c} \in \mathbb{Z}^d\}.\end{aligned}$$

□

Theorem 254. *Let $\mathcal{S}$ be a zero-SCC and let $q_{\mathtt{z}}$ be the zero-state of $\mathcal{S}$. Let $d = \dim(\mathcal{S})$.*

Each state of $\mathcal{S}$ has r^d incoming transitions from states in $\mathcal{S}$.

Proof. From Theorem 168, if $d = n$, $q_{\mathtt{z}}$ is the unique state in $\mathcal{S}$ and for all $\alpha \in \Sigma_r^n$, $\delta(q_{\mathtt{z}}, \alpha) = q_{\mathtt{z}}$. Therefore, there are r^n incoming transitions in $q_{\mathtt{z}}$.

From Theorem 169, if $d = 0$, then $q_{\mathtt{z}}$ is the unique state in $\mathcal{S}$ and there is only one incoming transition in $q_{\mathtt{z}}$ which is labeled by o.

Suppose now that $0 < d < n$. Let $q \in \mathcal{S}$ and let $v_{q_{\mathtt{z}}q} \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. From Theorem 172 and Lemma 253, there exists an integer matrix $\mathbf{B}$ of rank $n-d$ and a prime matrix $\mathbf{Y} \in \mathbb{N}^{n\times d}$ such that

$$L_{\mathcal{A}}(q_{\mathtt{z}} \to q) = \{v \mid \mathbf{B}(\langle ov\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\} \tag{7.57}$$

and

$$\begin{aligned}&\mathrm{aff}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)\}) \\ &\qquad = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{B}(\mathbf{x} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}\} \quad (7.58) \\ &\qquad = \{\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc} \mid \mathbf{c} \in \mathbb{Z}^d\} \quad (7.59)\end{aligned}$$

Since all elements in $\mathbf{Y}$ are non-negative integers, for all vectors $\mathbf{c} \in \{0, \ldots, r-1\}^d$, $\langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc} \geq \mathbf{0}$, and by definition of the encoding scheme, there exist $u \in (\Sigma_r^n)^*$ and $\alpha \in \Sigma_r^n$ such that $\langle ou\alpha\rangle_{r,n} = \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc}$. From (7.58) and (7.59), $\mathbf{B}(\langle u\alpha\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$ and from (7.57), $u\alpha \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$.

Let $\mathbf{c}_1, \mathbf{c}_2 \in \{0, \ldots, r-1\}^d$, $u_1, u_2 \in (\Sigma_r^n)^*$ and $\alpha_1, \alpha_2 \in \Sigma_r^n$ with $\langle ou_1\alpha_1\rangle_{r,n} = \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc}_1$ and $\langle ou_2\alpha_2\rangle_{r,n} = \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc}_2$. The following relation is immediate.

$$(\langle ou_1\alpha_1\rangle_{r,n} \equiv_r \langle ou_2\alpha_2\rangle_{r,n}) \Leftrightarrow (\mathbf{Yc}_1 \equiv_r \mathbf{Yc}_2) \tag{7.60}$$

By definition of the encoding scheme, $\langle ou_1\alpha_1\rangle_{r,n} \equiv \langle o\alpha_1\rangle_{r,n} \mod r$ and similarly, $\langle ou_2\alpha_2\rangle_{r,n} \equiv_r \langle o\alpha_2\rangle_{r,n}$. So, $(\langle o\alpha_1\rangle_{r,n} \equiv_r \langle o\alpha_2\rangle_{r,n}) \Leftrightarrow (\alpha_1 = \alpha_2)$, and (7.60) is equivalent to

$$(\alpha_1 = \alpha_2) \Leftrightarrow (\mathbf{Yc}_1 \equiv_r \mathbf{Yc}_2). \tag{7.61}$$

In addition, since $\mathbf{Y}$ is prime, according to Lemma 251, we have $\mathbf{Yc}_1 \equiv_r \mathbf{Yc}_2$ if and only if $\mathbf{c}_1 \equiv_r \mathbf{c}_2$, and since $\mathbf{c}_1, \mathbf{c}_2 \in \{0, \ldots, r-1\}^d$, by definition, $(\mathbf{c}_1 \equiv_r \mathbf{c}_2 \Leftrightarrow (\mathbf{c}_1 = \mathbf{c}_2)$. So, we have

$$\alpha_1 = \alpha_2 \Leftrightarrow \mathbf{c}_1 = \mathbf{c}_2 \tag{7.62}$$

Since there are r^d elements in $\{0, \ldots, r-1\}^d$, there are at least r^d different symbols α such that $\mathbf{B}(\langle ou\alpha\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$ for some $u \in (\Sigma_r^n)^*$, and since $u\alpha \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$ if $\mathbf{B}(\langle ouv\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$, there are at least r^d incoming transitions in q, each with a different label.

Let $\mathbf{c} \in \mathbb{Z}^d$ such that $\exists u \in (\Sigma_r^n)^*$, $\exists \alpha \in \Sigma_r^n$ with $\langle ou\alpha\rangle_{r,n} = \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc}$. From (7.57), (7.58) and (7.59), $\mathbf{B}(\langle ou\alpha\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$ and $u\alpha \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q)$. By definition, $\exists \mathbf{c}' \in \{0, \ldots, r-1\}^d$ such that $\mathbf{c}' = \mathbf{c} \mod r$. From above, $\exists u' \in (\Sigma_r^n)^*$, $\alpha' \in \Sigma_r^n$ with $\langle ou'\alpha'\rangle_{r,n} = \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n} + \mathbf{Yc}'$ and $\mathbf{B}(\langle ou'\alpha'\rangle_{r,n} - \langle ov_{q_{\mathtt{z}}q}\rangle_{r,n}) = \mathbf{0}$. Since $\mathbf{c}' = \mathbf{c} \mod r$ and since $\mathbf{Y}$ is prime, according to Lemma 251, $\mathbf{Yc} \equiv_r \mathbf{Yc}'$, and therefore, $\langle ou\alpha\rangle_{r,n} \equiv_r \langle ou'\alpha'\rangle_{r,n}$, i.e. $\alpha = \alpha'$. This implies that there are at most r^d different symbols labeling incoming transitions in q.

We conclude that there are r^d incoming transitions in q, each with a different label. □

7.8.2 Proof of Lemma 178

Lemma 255. *If $C \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$, for each face F of C, we have*

$$\mathrm{lin}_{\mathbb{Q}}(F) = \mathrm{lin}_{\mathbb{Q}}\big(\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n\big).$$

Proof. By definition, $\lin_{\mathbb{Q}}\left(\lin_{\mathbb{Q}}(F) \cap \mathbb{N}^n\right) \subseteq \lin_{\mathbb{Q}}(F)$. In addition, by definition, for all $\mathbf{x} \in \lin_{\mathbb{Q}}(F)$, there exists a finite set of elements $\mathbf{x}_1, \ldots, \mathbf{x}_k \in F$, such that $\mathbf{x}$ is a linear combination of $\mathbf{x}_1, \ldots, \mathbf{x}_k$. Since $\mathbf{x}_1, \ldots, \mathbf{x}_k \in \mathbb{Q}^n$, there exists $a \in \mathbb{N}$ such that $a\mathbf{x}_1, \ldots, a\mathbf{x}_k \in \mathbb{Z}^n$, and by definition of F, $a\mathbf{x}_1, \ldots, a\mathbf{x}_k \in F$. Since $F \subseteq C \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{x} \geq \mathbf{0}\}$, $a\mathbf{x}_1, \ldots, a\mathbf{x}_k \in F \cap \mathbb{N}^n$, and we conclude that $\lin_{\mathbb{Q}}(F) = \lin_{\mathbb{Q}}\left(\lin_{\mathbb{Q}}(F) \cap \mathbb{N}^n\right)$. □

7.8.3 Proof of Lemma 179

Lemma 256. *Let $q_{\mathtt{z}}$ be a zero-state of $\mathcal{S}$ and let $I_{\mathtt{p}} \subseteq \{1, \ldots, m\}$ be such that $i \in I_{\mathtt{p}}$ if and only if $\mathbf{a}_i.\mathbf{x} \leq b_i$ is pending in $q_{\mathtt{z}}$. Let $F = \{\mathbf{x} \in C \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}$.*

1. *For all $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $\langle u \rangle_{r,n} \in F$,*
2. *There exists a word $u \in (\Sigma_r^n)^*$ with $u \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that $\langle u \rangle_{r,n} \in F$ and for all proper faces F' of F, $\langle u \rangle_{r,n} \notin F'$,*
3. *$L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = \{v \in (\Sigma_r^n)^* \mid \langle ov \rangle_{r,n} \in \lin_{\mathbb{Q}}(F)\}$,*
4. *$\lin_{\mathbb{Q}}(\{\langle ov \rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}) = \lin_{\mathbb{Q}}(F)$*

Proof. From Lemma 160, there is a word $u_{\mathtt{z}} \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$ such that for all $i \in I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u_{\mathtt{z}} \rangle_{r,n} = 0$ and for all $i \in \{1, \ldots, m\} \setminus I_{\mathtt{p}}$, $\mathbf{a}_i.\langle u_{\mathtt{z}} \rangle_{r,n} < \min(b_i, -\|\mathbf{a}_i^+\|)$. Also, from Theorem 162, we have

$$L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}}) = \{u \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle ou \rangle_{r,n} = 0\}. \tag{7.63}$$

1. From Theorem 156, for all words u in $L_{\mathcal{A}}(q_{\mathtt{I}} \to q_{\mathtt{z}})$, $\langle u \rangle_{r,n} \in C \cap \mathbb{N}^n$, and by definition of $I_{\mathtt{p}}$, $\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\langle u \rangle_{r,n} = 0$. Therefore, by definition of F, $\langle u \rangle_{r,n} \in F$.
2. By definition, for all proper faces F' of F, $F' \subseteq \{\mathbf{x} \in F \mid \mathbf{a}.\mathbf{x} = 0\}$ for some inequation $\mathbf{a}.\mathbf{x} \leq b$ of $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ not pending in $q_{\mathtt{z}}$. So, by definition of $u_{\mathtt{z}}$, $u_{\mathtt{z}} \notin F'$.
3. By definition, $\lin_{\mathbb{Q}}(F) \subseteq \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}$, and therefore,

$$\lin_{\mathbb{Q}}(F) \cap \mathbb{N}^n \subseteq \{\mathbf{x} \in \mathbb{N}^n \mid \bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}. \tag{7.64}$$

Let $\mathbf{y} \in \mathbb{N}^n$ such that $\bigwedge_{i \in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{y} = 0$. We will prove that $\mathbf{y} \in \lin_{\mathbb{Q}}(F) \cap \mathbb{N}^n$.

For each $j \in \{1,\ldots,m\}\backslash I_{\mathtt{p}}$, let $a_j = \mathbf{a}_j.\mathbf{y}$. Let $a = 1+\max_{j\in\{1,\ldots,m\}\backslash I_{\mathtt{p}}}(|a_j|)$ and let $\mathbf{z} \in \mathbb{N}^n$ such that $\mathbf{z} = \mathbf{y} + a\langle u_{\mathbf{z}}\rangle_{r,n}$. By construction, $(\forall j \in \{1,\ldots,m\} \setminus I_{\mathtt{p}})(\mathbf{a}_j.\mathbf{z} < 0)$ and $(\forall i \in I_{\mathtt{p}})(\mathbf{a}_i.\mathbf{z} = 0)$, implying that $\mathbf{z} \in F$. Since $a\langle u_{\mathbf{z}}\rangle_{r,n} \in F$, we have that $\mathbf{y} \in \mathrm{lin}_{\mathbb{Q}}(F)$, and given that $\mathbf{y} \in \mathbb{N}^n$, $\mathbf{y} \in \mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n$. We have therefore proved that

$$\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n \supseteq \{\mathbf{x} \in \mathbb{N}^n \mid \bigwedge_{i\in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}. \tag{7.65}$$

Combining (7.64) and (7.65), we deduce that

$$\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n = \{\mathbf{x} \in \mathbb{N}^n \mid \bigwedge_{i\in I_{\mathtt{p}}} \mathbf{a}_i.\mathbf{x} = 0\}. \tag{7.66}$$

Finally, from (7.63) and (7.66), we conclude that

$$\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\} = \mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n. \tag{7.67}$$

4. From (7.67), we have

$$\mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}) = \mathrm{lin}_{\mathbb{Q}}\left(\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n\right). \tag{7.68}$$

From Lemma 178, $\mathrm{lin}_{\mathbb{Q}}(F) = \mathrm{lin}_{\mathbb{Q}}\left(\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n\right)$, and therefore,

$$\mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_{\mathtt{z}} \to q_{\mathtt{z}})\}) = \mathrm{lin}_{\mathbb{Q}}(F). \tag{7.69}$$

□

7.8.4 Proof of Lemma 182

Lemma 257. *For each face F of the characteristic cone C, there exists one and only one zero-state, denoted q_F, such that the encodings, possibly suffixed by a sequence of o symbols, of all integer elements in F which do not belong to a proper face of F label paths from $q_{\mathtt{I}}$ to q_F.*

In addition, the state q_F is such that

- $L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle ov\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)\}$, *and*
- $\mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) \supseteq \mathrm{lin}_{\mathbb{Q}}(F)$.

Proof. By definition, $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0}\}$ and

$$F = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0} \wedge \mathbf{A}'\mathbf{x} = \mathbf{0}\} \tag{7.70}$$

where $\mathbf{A}'\mathbf{x} \leq \mathbf{0}$ is a subsystem of $\mathbf{A}\mathbf{x} \leq \mathbf{0}$.

Recall that the inequations in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ are $\mathbf{a}_1.\mathbf{x} \leq b_1, \ldots, \mathbf{a}_m.\mathbf{x} \leq b_m$. Let $I_F \subseteq \{1, \ldots, m\}$ be the set of indices i such that $\mathbf{a}_i.\mathbf{x} \leq b_i$ is an implicit equation in $\mathbf{A}\mathbf{x} \leq \mathbf{0} \wedge \mathbf{A}'\mathbf{x} = \mathbf{0}$. By definition, for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\left(\bigwedge_{i \in I_F} \mathbf{a}_i.\mathbf{x} = 0\right) \Rightarrow \mathbf{A}'\mathbf{x} = \mathbf{0}\,. \tag{7.71}$$

Any proper face F' of F is such that

$$F' \subseteq \{\mathbf{x} \in F \mid \mathbf{a}_j.\mathbf{x} = 0\},$$

for some $j \in \{1, \ldots, m\} \setminus I_F$.

So, for all encodings u of elements which do not belong to a proper face of F, we have

$$\begin{array}{l} \mathbf{a}_i.\langle u \rangle_{r,n} = 0 \;\text{ if }\; i \in I_F \\ \mathbf{a}_i.\langle u \rangle_{r,n} < 0 \;\text{ if }\; i \in \{1, \ldots, m\} \setminus I_F\,. \end{array} \tag{7.72}$$

Thanks to Lemma 84, for all $k \geq |\min(\beta, -\|a^+\|)|$, we have

$$\begin{array}{ll} \mathbf{a}_i.\langle uo^k \rangle_{r,n} = 0 & \text{if }\; i \in I_F \\ \mathbf{a}_i.\langle uo^k \rangle_{r,n} \leq \min(b_1, -\|a^+\|) & \text{if }\; i \in \{1, \ldots, m\} \setminus I_F\,. \end{array} \tag{7.73}$$

So, we have

$$\begin{aligned} uo^k \div L(\mathcal{A}) &= \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in \{1,\ldots,m\}} \mathbf{a}_i.\langle uo^k w \rangle_{r,n} \leq b_i\} && (7.74) \\ &= \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle uo^k w \rangle_{r,n} \leq b_i\} && (7.75) \\ &= \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i\} && (7.76) \end{aligned}$$

Similarly, we have

$$\begin{aligned} uo^{k+1} \div L(\mathcal{A}) &= \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in \{1,\ldots,m\}} \mathbf{a}_i.\langle uo^{k+1} w \rangle_{r,n} \leq b_i\} && (7.77) \\ &= \{w \in (\Sigma_r^n)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle ow \rangle_{r,n} \leq b_i\} && (7.78) \end{aligned}$$

Since $\{w \in (\Sigma_r^n)^* \mid \bigwedge_{i\in I_F} \mathbf{a}_i.\langle ow\rangle_{r,n} \leq b_i\} \neq \emptyset$, there exists a state Q_F such that $uo^k \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_F)$, and since $uo^k \div L(\mathcal{A}) = uo^{k+1}L(\mathcal{A})$, $\delta(q_F, o) = q_F$ and q_F is a zero-state.

Since F is a cone, thanks to Lemma 28, from (7.70) and (7.71), we deduce

$$\mathrm{lin}_{\mathbb{Q}}(F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge_{i\in I_F} \mathbf{a}_i.\mathbf{x} = 0\}\,. \tag{7.79}$$

From (7.76) and (7.79), we deduce that for all $\langle ov\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)$, $v \in L_{\mathcal{A}}(q_F \to q_F)$, i.e.

$$L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle ov\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)\} \tag{7.80}$$

Finally, thanks to Lemma 178, $\mathrm{lin}_{\mathbb{Q}}(F) = \mathrm{lin}_{\mathbb{Q}}\left(\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n\right)$, and thanks to (7.80), we have

$$\begin{array}{rcl} \mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) & \supseteq & \mathrm{lin}_{\mathbb{Q}}(\{\langle ov\rangle_{r,n} \mid \langle ov\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)\}) \\ & \supseteq & \mathrm{lin}_{\mathbb{Q}}(\mathrm{lin}_{\mathbb{Q}}(F) \cap \mathbb{N}^n) \\ & \supseteq & \mathrm{lin}_{\mathbb{Q}}(F)\,. \end{array}$$

□

7.8.5 Proof of Lemma 217

Lemma 258. *For each face F of C,* $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F) = \mathrm{aff}_{\mathbb{Q}}\left(\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n}+F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\right)$.

Proof. By definition, $\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \supseteq \mathrm{aff}_{\mathbb{Q}}\left(\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\right)$.

In addition, by definition, for all $\mathbf{x} \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)$, there exists a finite set of elements $\mathbf{x}_1, \ldots, \mathbf{x}_k \in \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$, such that $\mathbf{x} = \sum_{i=1}^k a_i\mathbf{x}_i$ with $\sum_{i=1}^k a_i = 1$. Therefore, $\mathbf{x} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n} = \sum_{i=1}^k a_i(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n})$. By definition, $\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n} \in F$, for all $i \in \{1, \ldots, k\}$, and so we have $\mathbf{x} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n} \in \mathrm{lin}_{\mathbb{Q}}(F)$.

Since $\mathbf{x}_1, \ldots, \mathbf{x}_k \in \mathbb{Q}^n$, there exists $a \in \mathbb{N}$ such that $a\mathbf{x}_1, \ldots, a\mathbf{x}_k \in \mathbb{Z}^n$, and since F is the face of a cone, it is itself a cone, and by definition, $a(\mathbf{x}_1 - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}), \ldots, a(\mathbf{x}_k - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) \in F \cap \mathbb{Z}^n$. So, by definition, $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_1 - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}), \ldots, \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_k - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) \in \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F \cap \mathbb{Z}^n$. Thanks to Lemma 193, $\mathrm{sign}_r\left(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n})\right) = \alpha_{\mathtt{sign}}$, for all $i \in \{1, \ldots, k\}$ and therefore, $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n}) \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid$

$\mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$. Also, by definition, $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$. So, by construction, we have

$$\begin{aligned}
\mathbf{x} &= \frac{1}{a}\left(\sum_{i=1}^{k} a_i \cdot a\mathbf{x}_i\right) \\
&= \frac{1}{a}\left(a\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + \sum_{i=1}^{k} a_i \cdot a(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n})\right) \\
&= \frac{1}{a}\left(a\langle\alpha_{\mathtt{sign}}\rangle_{r,n} - \langle\alpha_{\mathtt{sign}}\rangle_{r,n} + \sum_{i=1}^{k} a_i(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n})\right) \\
&= \frac{1}{a}\left((a-1)\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + \sum_{i=1}^{k} a_i(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + a(\mathbf{x}_i - \langle\alpha_{\mathtt{sign}}\rangle_{r,n})\right) \\
&= \frac{1}{a}\left((a-1)\mathbf{y}_0 + \sum_{i=1}^{k} a_i\mathbf{y}_i\right)
\end{aligned}$$

with $\mathbf{y}_0, \ldots, \mathbf{y}_k \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}$. Also, $\frac{1}{a}(a - 1 + \sum_{i=1}^{k} a_i) = 1$. So, $\mathbf{x} \in \mathrm{aff}_{\mathbb{Q}}\left(\mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\right)$. □

7.8.6 Proof of Lemma 221

Lemma 259. *For each face F of the characteristic cone C, there exists one and only one sign-state, denoted q_F, such that the encodings, possibly suffixed by a sequence of $\alpha_{\mathtt{sign}}$ symbols, of all integer elements in $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ which do not belong to a proper face of $\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F$ label paths from $q_{\mathtt{I}}$ to q_F.*

In addition, the state q_F is such that

- $L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)\}$, *and*
- $\mathrm{aff}_{\mathbb{Q}}(\{\langle\alpha_{\mathtt{sign}}v\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) \supseteq \mathrm{aff}_{\mathbb{Q}}(\langle\alpha_{\mathtt{sign}}\rangle_{r,n} + F)$.

Proof. By definition, $C = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0}\}$ and

$$F = \{\mathbf{x} \in \mathbb{Q}^n \mid \mathbf{A}\mathbf{x} \leq \mathbf{0} \wedge \mathbf{A}'\mathbf{x} = \mathbf{0}\} \tag{7.81}$$

where $\mathbf{A}'\mathbf{x} \leq \mathbf{0}$ is a subsystem of $\mathbf{A}\mathbf{x} \leq \mathbf{0}$.

Recall that the inequations in $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ are $\mathbf{a}_1.\mathbf{x} \leq b_1, \ldots, \mathbf{a}_m.\mathbf{x} \leq b_m$. Let $I_F \subseteq \{1, \ldots, m\}$ be the set of indices i such that for all $\mathbf{x} \in \mathbb{Q}^n$, $\mathbf{x} \in F \Rightarrow \mathbf{a}_i.\mathbf{x} =$

0. By definition, for all $\mathbf{x} \in \mathbb{Q}^n$, we have

$$\left(\bigwedge_{i \in I_F} \mathbf{a}_i.\mathbf{x} = 0 \right) \Rightarrow \mathbf{A}'\mathbf{x} = \mathbf{0}\,. \tag{7.82}$$

Any proper face F' of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F$ is such that

$$F' \subseteq \{\mathbf{x} \in \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F \mid \mathbf{a}_j.(\mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = 0\},$$

for some $j \in \{1, \ldots, m\} \setminus I_F$.

So, for all encodings u of elements of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F$ which do not belong to a proper face of $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + F$, we have

$$\begin{array}{l} \mathbf{a}_i.(\langle u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) = 0 \;\text{ if }\; i \in I_F \\ \mathbf{a}_i.(\langle u \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}) < 0 \;\text{ if }\; i \in \{1, \ldots, m\} \setminus I_F\,. \end{array} \tag{7.83}$$

By definition of the encoding scheme, we have

$$\langle u\alpha_{\mathtt{sign}} \rangle_{r,n} = r\langle u \rangle_{r,n} + \langle o\alpha_{\mathtt{sign}} \rangle_{r,n} = r\langle u \rangle_{r,n} + (1-r)\langle \alpha_{\mathtt{sign}} \rangle_{r,n}.$$

Therefore, we deduce that

$$\begin{array}{l} \mathbf{a}_i.\langle u\alpha_{\mathtt{sign}} \rangle_{r,n} = \mathbf{a}_i.\langle \alpha_{\mathtt{sign}} \rangle_{r,n} \;\text{ if }\; i \in I_F \\ \mathbf{a}_i.\langle u\alpha_{\mathtt{sign}} \rangle_{r,n} < \mathbf{a}_i.\langle \alpha_{\mathtt{sign}} \rangle_{r,n} \;\text{ if }\; i \in \{1, \ldots, m\} \setminus I_F\,. \end{array} \tag{7.84}$$

Thanks to Lemma 84, for all $k \geq |\min(\beta, -\| a^+ \|)|$, we have

$$\begin{array}{ll} \mathbf{a}_i.\langle u\alpha^k_{\mathtt{sign}} \rangle_{r,n} = 0 & \text{if } i \in I_F \\ \mathbf{a}_i.\langle u\alpha^k_{\mathtt{sign}} \rangle_{r,n} \leq \min(b_1, -\| a^+ \|) & \text{if } i \in \{1, \ldots, m\} \setminus I_F\,. \end{array} \tag{7.85}$$

So, we have

$$\begin{aligned} u\alpha^k_{\mathtt{sign}} \div L(\mathcal{A}) &= \{w \in (\Sigma^n_r)^* \mid \bigwedge_{i \in \{1,\ldots,m\}} \mathbf{a}_i.\langle u\alpha^k_{\mathtt{sign}} w \rangle_{r,n} \leq b_i\} && (7.86) \\ &= \{w \in (\Sigma^n_r)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle u\alpha^k_{\mathtt{sign}} w \rangle_{r,n} \leq b_i\} && (7.87) \\ &= \{w \in (\Sigma^n_r)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle \alpha_{\mathtt{sign}} w \rangle_{r,n} \leq b_i\} && (7.88) \end{aligned}$$

Similarly, we have

$$\begin{aligned} u\alpha^{k+1}_{\mathtt{sign}} \div L(\mathcal{A}) &= \{w \in (\Sigma^n_r)^* \mid \bigwedge_{i \in \{1,\ldots,m\}} \mathbf{a}_i.\langle u\alpha^{k+1}_{\mathtt{sign}} w \rangle_{r,n} \leq b_i\} && (7.89) \\ &= \{w \in (\Sigma^n_r)^* \mid \bigwedge_{i \in I_F} \mathbf{a}_i.\langle \alpha_{\mathtt{sign}} w \rangle_{r,n} \leq b_i\} && (7.90) \end{aligned}$$

Since $\{w \in (\Sigma_r^n)^* \mid \bigwedge_{i\in I_F} \mathbf{a}_i.\langle \alpha_{\mathtt{sign}} w\rangle_{r,n} \leq b_i\} \neq \emptyset$, there exists a state q_F such that $u\alpha_{\mathtt{sign}}^k \in L_{\mathcal{A}}(q_{\mathtt{I}} \to q_F)$, and since $u\alpha_{\mathtt{sign}}^k \div L(\mathcal{A}) = u\alpha_{\mathtt{sign}}^{k+1} L(\mathcal{A})$, $\delta(q_F, \alpha_{\mathtt{sign}}) = q_F$ and q_F is a sign-state.

Since F is a cone, thanks to Lemma 215, from (7.81) and (7.82), we deduce

$$\mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F) = \{\mathbf{x} \in \mathbb{Q}^n \mid \bigwedge_{i\in I_F} \mathbf{a}_i.(\mathbf{x} - \langle \alpha_{\mathtt{sign}}\rangle_{r,n}) = 0\}\,. \tag{7.91}$$

From (7.88) and (7.91), we deduce that for all $\langle \alpha_{\mathtt{sign}} v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F)$, $v \in L_{\mathcal{A}}(q_F \to q_F)$, i.e.

$$L_{\mathcal{A}}(q_F \to q_F) \supseteq \{v \in (\Sigma_r^n)^* \mid \langle \alpha_{\mathtt{sign}} v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F)\} \tag{7.92}$$

Finally, thanks to Lemma 217, $\mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F) = \mathrm{aff}_{\mathbb{Q}}\left(\mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\right)$, and thanks to (7.92), we have

$$\begin{aligned}
&\mathrm{aff}_{\mathbb{Q}}(\{\langle \alpha_{\mathtt{sign}} v\rangle_{r,n} \mid v \in L_{\mathcal{A}}(q_F \to q_F)\}) \\
&\qquad\supseteq\ \mathrm{aff}_{\mathbb{Q}}(\{\langle \alpha_{\mathtt{sign}} v\rangle_{r,n} \mid \langle \alpha_{\mathtt{sign}} v\rangle_{r,n} \in \mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F)\}) \\
&\qquad\supseteq\ \mathrm{aff}_{\mathbb{Q}}\left(\mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F) \cap \{\mathbf{x} \in \mathbb{Z}^n \mid \mathrm{sign}_r(\mathbf{x}) = \alpha_{\mathtt{sign}}\}\right) \\
&\qquad\supseteq\ \mathrm{aff}_{\mathbb{Q}}(\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + F)\,.
\end{aligned}$$

□

7.8.7 Proof of Theorem 243

Theorem 260. *For each $q \in Q$, let (X_q, Y_q) be an extended Hilbert basis generating $S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)$, and let Q_C be the set of states q such that there is a path from q to a sign-state labeled by a sequence of $\alpha_{\mathtt{sign}}$ symbols.*

Let $X'_{q_{\alpha_{\mathtt{sign}}}} = \left\{r \cdot \mathbf{x} + \langle o\alpha\rangle_{r,n} \mid \exists q \in Q, \alpha \in \Sigma_r^n \left(\delta(q,\alpha) = q_{\alpha_{\mathtt{sign}}} \wedge \mathbf{x} \in X_q\right)\right\}$ and let $q_{\alpha_{\mathtt{sign}}} \in Q$ such that $\delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}}) = q_{\alpha_{\mathtt{sign}}}$.

The set $-\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q\in Q_C} X_q\right)$ is an Hilbert basis generating $C \cap \mathbb{Z}^n$.

Proof. We show that $\mathrm{cone}_{\mathbb{Z}}\left(-\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q\in Q_C} X_q\right)\right) = C \cap \mathbb{Z}^n$ by proving the mutual inclusion.

- Thanks to Theorem 234, we have

$$C \cap \mathbb{Z}^n = -\langle \alpha_{\mathtt{sign}}\rangle_{r,n} + \cup_{q\in Q_C}\left(S_{\mathcal{A}}^q + (C \cap \mathbb{Z}^n)\right). \tag{7.93}$$

By definition, $X_q \subseteq S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$. Also, $q_{\alpha_{\mathtt{sign}}} \in Q_C$ and $X'_{q_{\alpha_{\mathtt{sign}}}} \subseteq S^{q_{\alpha_{\mathtt{sign}}}}_{\mathcal{A}}$. So, we have

$$-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q \right) \subseteq C \cap \mathbb{Z}^n. \tag{7.94}$$

Since C is a cone, by definition, we have

$$\mathrm{cone}_{\mathbb{Z}} \left(-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q \right) \right) \subseteq C \cap \mathbb{Z}^n. \tag{7.95}$$

- Let $Y_{\min}$ be the minimal Hilbert basis generating $C \cap \mathbb{Z}^n$ and let $\mathbf{y} \in Y_{\min}$.

  Since C is pointed, thanks to Lemma 40, for each $q \in Q$, there exists a minimal extended Hilbert basis $(X_{q,\min}, Y_{\min})$ generating the set $S^q_{\mathcal{A}} + (C \cap \mathbb{Z}^n)$ with $X_{q,\min} \subseteq X_q$. Thanks to Theorem 240, for each $\mathbf{x} \in X_{q,\min}$, there exists an encoding u of $\mathbf{x}$ labeling an acyclic path from $q_{\mathtt{I}}$ to q, and since $\mathcal{A}$ is reduced minimal, u is the minimal encoding of $\mathbf{x}$.

  Let $\mathbf{z}$ such that

  $$\mathbf{z} = \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \mathbf{y}. \tag{7.96}$$

  Thanks to Theorem 199, there exists a state $q_{\mathbf{z}} \in Q_C$ such that $\mathbf{z} \in S^{q_{\mathbf{z}}}_{\mathcal{A}}$. By definition, $\mathbf{z} = \mathbf{x} + \mathbf{y}'$ for some $\mathbf{x} \in X_{q_{\mathbf{z}},\min}$ and $\mathbf{y}' \in C \cap \mathbb{Z}^n$. There are two possibilities, either $\mathbf{x} = \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$ or $\mathbf{x} = \mathbf{z}$. Indeed, otherwise, one would have $\mathbf{y} = \mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \mathbf{y}'$ with both $\mathbf{x} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$ and $\mathbf{y}'$ in $(C \cap \mathbb{Z}^n) \setminus \{\mathbf{0}\}$, and therefore, $\mathbf{y}$ would not be in the minimal Hilbert basis generating $C \cap \mathbb{Z}^n$.

  - Suppose that $\mathbf{x} = \mathbf{z}$. Then we have

    $$\mathbf{y} \in -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + X_{q_{\mathbf{z}}}. \tag{7.97}$$

  - Suppose that $\mathbf{x} = \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$. By definition, the minimal encoding of $\mathbf{x}$ is $\alpha_{\mathtt{sign}}$ and thanks to Theorem 240, $q_{\mathbf{z}} = \delta(q_{\mathtt{I}}, \alpha_{\mathtt{sign}})$, and by definition, $q_{\mathbf{z}} = q_{\alpha_{\mathtt{sign}}}$. Let u be the minimal encoding of $\mathbf{z}$. We have

    $$\langle u \rangle_{r,n} = \mathbf{z}. \tag{7.98}$$

    We prove that there exist a state $q' \in Q$ and a symbol $\alpha \in \Sigma^n_r$ with $\delta(q', \alpha) = q_{\alpha_{\mathtt{sign}}}$ such that $\mathbf{z} = r \cdot \mathbf{z}' + \langle o\alpha \rangle_{r,n}$ for some $\mathbf{z}' \in X_{q',\min}$, and this implies that

    $$\mathbf{y} \in -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + X'_{q_{\alpha_{\mathtt{sign}}}}. \tag{7.99}$$

* Suppose that $|u| = 1$. This case is not possible. Indeed, since $\mathbf{y}$ is in the minimal Hilbert basis of $C \cap \mathbb{Z}^n$, $\mathbf{y} \neq \mathbf{0}$, and therefore, by construction, $\mathbf{z} \neq \langle \alpha_{\mathtt{sign}} \rangle_{r,n}$, i.e. $u \neq \alpha_{\mathtt{sign}}$ and $|u| > 1$.
* Suppose that $|u| = 2$. Then, by definition, $u = \alpha_{\mathtt{sign}}\alpha$ and $\delta(q_{\alpha_{\mathtt{sign}}}, \alpha) = q_{\alpha_{\mathtt{sign}}}$. Also, by definition of the encoding scheme, $\langle u \rangle_{r,n} = r \cdot \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \langle o\alpha \rangle_{r,n}$, and from above, $\langle \alpha_{\mathtt{sign}} \rangle_{r,n} \in X_{q_{\alpha_{\mathtt{sign}}},\min}$.
* Suppose that $|u| \geq 3$. Then, by definition, $u = \alpha_{\mathtt{sign}} v \alpha$ for some $v \in (\Sigma_r^n)^*$. Let $q \in Q$ such that $\hat{\delta}(q_{\mathtt{I}}, \alpha_{\mathtt{sign}} v) = q$. Since u is a minimal encoding, $\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} \neq \langle u \rangle_{r,n}$ and by definition of the encoding scheme,

$$\langle u \rangle_{r,n} = r \cdot \langle \alpha_{\mathtt{sign}} v \rangle_{r,n} + \langle o\alpha \rangle_{r,n}. \tag{7.100}$$

By definition,

$$\langle \alpha_{\mathtt{sign}} v \rangle_{r,n} = \mathbf{x}_v + \mathbf{y}_v \,, \tag{7.101}$$

with $\mathbf{x}_v \in X_{q,\min}$ and $\mathbf{y}_v \in C \cap \mathbb{Z}^n$. Thanks to Theorem 240, the minimal encoding of $\mathbf{x}_v$ labels a path from $q_{\mathtt{I}}$ to q and thanks to Lemma 192, the sign symbol of the minimal encoding is $\alpha_{\mathtt{sign}}$. Let $\alpha_{\mathtt{sign}} w$ be the minimal encoding of $\mathbf{x}_v$. By construction and by definition of the encoding scheme, we have

$$\begin{aligned} \mathbf{y} &= -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + r \cdot (\mathbf{x}_v + \mathbf{y}_v) + \langle o\alpha \rangle_{r,n} \\ &= -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + r \cdot \langle \alpha_{\mathtt{sign}} w \rangle_{r,n} + \langle o\alpha \rangle_{r,n} + r \cdot \mathbf{y}_v \\ &= -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \langle \alpha_{\mathtt{sign}} w\alpha \rangle_{r,n} + r \cdot \mathbf{y}_v \end{aligned}$$

By construction, $\alpha_{\mathtt{sign}} w\alpha$ labels a path from $q_{\mathtt{I}}$ to $q_{\alpha_{\mathtt{sign}}} \in Q_C$, and therefore, thanks to Theorem 199, $\langle \alpha_{\mathtt{sign}} w\alpha \rangle_{r,n} \in \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + (C \cap \mathbb{Z}^n)$, i.e. $\langle \alpha_{\mathtt{sign}} w\alpha \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n} \in C \cap \mathbb{Z}^n$. Since C is pointed, $\langle \alpha_{\mathtt{sign}} w\alpha \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \mathbf{y}_v \neq \mathbf{0}$. So, since $\mathbf{y}$ is in the minimal Hilbert basis of $C \cap \mathbb{Z}^n$, we have

$$\mathbf{y}_v = \mathbf{0}. \tag{7.102}$$

Indeed, otherwise, $\mathbf{y} = \mathbf{y}_1 + \mathbf{y}_2$, with $\mathbf{y}_1 = \langle \alpha_{\mathtt{sign}} w\alpha \rangle_{r,n} - \langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \mathbf{y}_v \neq \mathbf{0}$ and $\mathbf{y}_2 = (r-1) \cdot \mathbf{y}_v \neq \mathbf{0}$, violating the definition of the minimal Hilbert basis. From (7.102), since $\mathbf{y}_v = \mathbf{0}$ and $\mathbf{y} = -\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \langle u \rangle_{r,n}$, we deduce that

$$\langle u \rangle_{r,n} = r \cdot \mathbf{x}_v + \langle o\alpha \rangle_{r,n} \,, \tag{7.103}$$

with $\mathbf{x}_v \in X_{q,\min} \subseteq X_q$, $q \in Q_C$ and $\delta(q,\alpha) = q_{\alpha_{\mathtt{sign}}}$.

From (7.97) and (7.99), we deduce that for all $\mathbf{y}$ in the minimal Hilbert basis $Y_{\min}$ generating $C \cap \mathbb{Z}^n$, we have

$$\mathbf{y} \in \left(-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q \right) \right) \tag{7.104}$$

By definition, $\mathrm{cone}_{\mathbb{Z}}(Y_{\min}) = C \cap \mathbb{Z}^n$, and we have that

$$\mathrm{cone}_{\mathbb{Z}} \left(-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q \right) \right) \subseteq C \cap \mathbb{Z}^n. \tag{7.105}$$

From (7.94) and (7.106), we conclude that

$$\mathrm{cone}_{\mathbb{Z}} \left(-\langle \alpha_{\mathtt{sign}} \rangle_{r,n} + \left(X'_{q_{\alpha_{\mathtt{sign}}}} \cup \bigcup_{q \in Q_C} X_q \right) \right) = C \cap \mathbb{Z}^n. \tag{7.106}$$

□

Chapter 8

General Conclusion

8.1 Summary

The main results of this thesis are new algorithms for extracting information about Presburger-definable sets represented by Number Decision Diagrams (NDDs). The extracted information is a quantifier-free Presburger formula (or a set of generators) corresponding to the represented set or an over-approximation of this set.

In Chapter 6, we have presented two algorithms, QAFFINEHULL and ZAFFINEHULL, that take as input a reduced (non-deterministic) NDD $\mathcal{A}$ using the synchronous encoding scheme and compute, in polynomial time, the affine hulls over $\mathbb{Q}$ and over $\mathbb{Z}$ respectively of the set represented by $\mathcal{A}$.

The restrictions that the NDDs be reduced and use the synchronous encoding scheme are not constraining. Indeed, any NDD can be reduced in linear time, and similarly, any NDD using the reverse synchronous encoding scheme can be converted into an NDD using the synchronous encoding scheme representing the same set in linear time.

We conclude that computing the affine hull, whether over $\mathbb{Q}$ or $\mathbb{Z}$, presents the advantages of being fast and of generating simple representations, i.e. a conjunction of linear equations (and congruences over $\mathbb{Z}$) and a set of at most n generators, where n is the number of components of the vectors. However, formulas corresponding exactly to the represented sets might be required.

In Chapter 7, we have presented an algorithm GENERATEFORMULA computing a formula corresponding exactly to the set represented by a reduced minimal NDD $\mathcal{A}$ using the synchronous encoding scheme, provided that the represented set is the set of integer elements of a convex polyhedron P, i.e. the integer solutions of a conjunction of linear inequations. The algorithm exploits the property

that P is decomposable into $P = Q + C$, where C is the characteristic cone of P and Q is a polytope. When dealing with the integer elements in P, i.e. $P \cap \mathbb{Z}^n$, this property translated into the existence of a pair of finite sets $(S_{\mathtt{cst}}, S_{\mathtt{per}})$, where $S_{\mathtt{per}}$ is an Hilbert basis, such that the positive integer combinations of elements in those sets correspond exactly to $P \cap \mathbb{Z}^n$, more precisely

$$P \cap \mathbb{Z}^n = \bigcup_{\mathbf{x}_i \in S_{\mathtt{cst}}} \{\mathbf{x}_i + \sum_{\mathbf{y}_j \in S_{\mathtt{per}}} a_j \cdot \mathbf{y}_j \mid a_i \in \mathbb{N}\}.$$

Two important components of the algorithm COMPUTEFORMULA are the functions CHARCONEFORMULA and COMPUTEBASIS.

- The function CHARCONEFORMULA, relying on detailed structural properties of the NDD, generates a system of homogeneous inequations $\mathbf{Cx} \leq \mathbf{0}$ corresponding to the characteristic cone of P in polynomial time.
- The function COMPUTEBASIS computes a pair of finite set $(S_{\mathtt{cst}}, S_{\mathtt{per}})$ generating $P \cap \mathbb{Z}^n$. In the worst-case, the number of elements in the set $S_{\mathtt{cst}}$ is proportional to the number of acyclic paths in the NDD and this lead to an exponential worst-case complexity.

Combining those results, the formula generated by COMPUTEFORMULA, corresponding exactly to the set $P \cap \mathbb{Z}^n$, is

$$\bigvee_{\mathbf{x}_i \in S_{\mathtt{cst}}} \mathbf{C}(\mathbf{x} - \mathbf{x}_i) \leq \mathbf{0}.$$

Interestingly, thanks to [Kla04b], this type of quantifier-free Presburger formula can be converted back into NDDs in polynomial time, as recalled in Section 5.4.

Note that the functions CHARCONEFORMULA and COMPUTEBASIS handle NDDs accepting encodings with the same sign. In order to deal with the case of multiple signs, one has to perform the computation for each sign and merge the results.

The overall algorithm has been tested with a prototype implementation, and the experimental results are very encouraging : the generation of formulas and bases corresponding to NDDs with more than 100,000 states can be achieved in seconds. Experimental results suggest that the actual cost is proportional to the size of the NDD as well as to both the number of elements in the basis and their encoding lengths, and those are in general much smaller compared to the number of acyclic paths. This explains why our algorithm performs well in practice despite an exponential worst-case complexity.

8.2 Discussion

We briefly recall the main elements presented in Sections 6.5.1 and 7.7.1.

An algorithm computing the affine hull over $\mathbb{Q}$ of the set represented by an NDD $\mathcal{A}$ can be deduced from [MS04] in which a method for computing the affine relations holding at control locations in affine programs is given. The time complexity of this adapted algorithm is $\mathcal{O}(|\Delta| \cdot n^3)$, where Δ is the transition relation of $\mathcal{A}$ and n is the number of components of the vectors in the represented set.

The problem of computing the affine hull over $\mathbb{Q}$ of a set represented by an NDD (when restricting to positive integer elements) has been also addressed in [Ler04a]. The time complexity of the algorithm in [Ler04a] is $\mathcal{O}(|\Delta| \cdot n^4)$. So, our main contribution with respect to computing affine hull over $\mathbb{Q}$ is the presentation of an algorithm of better time complexity than existing work. Indeed, the time complexity of our algorithm QAFFINEHULL is $\mathcal{O}(|\Delta| \cdot n)$ (or $\mathcal{O}(|\Delta| \cdot n^2)$ if one requires at most n generators or a set of equations).

An algorithm computing the affine hull over $\mathbb{Z}$ of a set represented by an NDD can be deduced from [Gra91] which presents an algorithm for computing the affine relations as well as the linear congruences holding at some control locations in affine programs. Although the algorithm always terminates, the number of execution steps is not bounded. By using modular arithmetic, our algorithm ZAFFINEHULL computes the affine hull over $\mathbb{Z}$ in polynomial time. This was made possible by exploiting specific properties of NDDs.

In [Ler03], an algorithm computing another over-approximation, the *semi-affine hull*, has been presented. The semi-affine hull S' of a set S is the smallest union of affine spaces over $\mathbb{Q}$ including S, and so, $S \subseteq S' \subseteq \mathrm{aff}_{\mathbb{Q}}(S)$, i.e., this over-approximation is closer to the original set. The number of affine spaces in a semi-affine space is not bounded, and in particular, the semi-affine hull of a finite set is the set itself. Therefore, given an NDD representing a finite set, the size of the semi-affine hull is proportional to the number of acyclic paths, i.e., exponential in the number of states. This explains the exponential complexity of the algorithm computing semi-affine hulls presented in [Ler03].

An algorithm computing the convex hull over $\mathbb{Q}$ has been presented in [FL05]. The time complexity is also exponential and the practical costs are not clear.

There is also other work on the computation of formulas corresponding exactly to sets represented by NDDs. In [Lug04], an algorithm has been proposed which computes the semi-linear set corresponding to the set $S_{\mathcal{A}}$ represented by an NDD

$\mathcal{A}$, with the restriction that

$$S_{\mathcal{A}} = \bigcup_{\mathbf{x}_i \in S_{\mathtt{cst}}} \{\mathbf{x}_i + \sum_{\mathbf{y}_j \in S_{\mathtt{per}}} a_j \cdot \mathbf{y}_j \mid a_i \in \mathbb{N}\}.$$

for some finite sets of vectors $S_{\mathtt{cst}}$ and $S_{\mathtt{per}}$. Note that $S_{\mathcal{A}}$ is not necessarily the set of integer elements of a polyhedron since $S_{\mathtt{per}}$ is not restricted to be an Hilbert basis.

Interestingly, the algorithm does not depend on the implementation details associated to NDDs, such as the encoding scheme, however its computational cost (double exponential worst case complexity) prevents its use in practice.

More recently, [Ler04b, Ler05] detail a polynomial algorithm computing a Presburger formula corresponding to the set represented by a deterministic NDD using the reverse synchronous encoding scheme given as input. The sole restriction on the input NDD is that it must represent a subset of $\mathbb{N}^n$ for some $n \in \mathbb{N}$. Although the complexity of the algorithm is polynomial, the practical cost are significant. Indeed, the computational cost is at least $|Q|^4$, where Q is the set of states, and this leads to huge numbers when considering NDDs with thousands of states. Another aspect which may prevent the practical applications of the algorithm is the presence of a polynomial number of quantifiers in the generated formula.

8.3 Future Work

The general problem of extracting information from NDDs in a practical way is far from closed.

Although already discussed in [FL05], we think that more can be done with respect to the computation of the convex hull over $\mathbb{Q}$ of sets represented by NDDs. Given that the convex hull is always a polyhedron, it might appear interesting to adapt the algorithm of [CH78], in a way similar to what we have done regarding the algorithm of [MS04], and modify the widening criterion based on the properties of the encoding scheme. The general idea would be to compute for each state q the smallest polyhedron P_q such that for all encoding u labeling a path from an initial state to q, we have $\langle u \rangle_{r,n} \in P_q$. One could also consider the convex hull over $\mathbb{Z}$.

Also, the problem of directly generating a formula corresponding exactly to the set represented by an NDD is still open when using the synchronous encoding scheme. We think it might be interesting to tackle some restricted classes before

considering the general problem. In particular, minimal NDDs corresponding to integer elements satisfying a Boolean combination of inequations present interesting similarities with those corresponding to convex polyhedra. The concepts of sign-state, sign-loop, sign-SCC, representative matrix are unchanged, and this might be inferred from the fact that minimal NDDs representing integer elements of polyhedra are permutation-free as proved in Lemma 86. The concept of pending inequations defined as inequations that constrain the language accepted from a sign-state can be adapted naturally. The major difference is that one can not generalize easily the notion of characteristic cone and distinguish a single infinite part for the whole set. However, it can be proved from the construction algorithm given in Section 5.4 that, in any minimal NDD using the synchronous encoding scheme representing the integer solutions of a Boolean combination of inequations, the set of encodings labeling paths from the initial state to some state q corresponds also to the integer elements satisfying a Boolean combination of inequations. Our intuition is that one could compute for each state q a formula $\varphi_q(\mathbf{x})$ whose integer solutions are the vectors whose encodings label paths from the initial state to the state q and such that φ_q is a Boolean combination of inequations $\mathbf{a}_i.\mathbf{x} \leq b$ where the vectors of coefficients $\mathbf{a}_i$ could be computed from representative matrices of the sign-SCCs which are crossed when following paths from the initial state and q.

Another restricted class that might be of interest is the class of sets that can be represented by a conjunction of inequations and congruence relations. When restricting to encodings having one particular sign symbol, [Lug04] proved that a set S in this class can be generated from a pair of finite sets $(S_{\mathtt{cst}}, S_{\mathtt{per}})$, in a way similar to what is done in the case of convex polyhedra. Our intuition is that the properties and the algorithms presented in Chapter 7 can be adapted when considering NDDs representing integer solutions of systems of inequations and congruence relations. For example, sign-loop could have a length greater than one and the representative matrix should probably be a representative system of equations and congruence relations. Also, when dealing with integer elements in polyhedra, we associated a vector space to some strongly connected components of the NDD. The generalization could lead to associate $\mathbb{Z}$-modules rather than vector spaces to strongly connected components.

Finally, our work on the structure of NDDs representing the integer elements of polyhedra, and in particular the association of vector spaces over $\mathbb{Q}$ with some strongly connected components, has shown that one might use formulas to represent at least part of an NDD, raising the possibility of a more compact representa-

tion for representing sets of integer vectors.

Bibliography

[AC95] F. Ajili and E. Contejean. Complete solving of linear diophantine equations and inequations without adding variables. In *Principles and Practice of Constraint Programming*, pages 1–17, 1995.

[AC97] F. Ajili and E. Contejean. Avoiding slack variables in the solving of linear Diophantine equations and inequations. *Theoretical Computer Science*, 173(1):183–208, February 1997.

[BC96] A. Boudet and H. Comon. Diophantine equations, Presburger arithmetic and finite automata. In *Proceedings of CAAP'96*, number 1059 in Lecture Notes in Computer Science, pages 30–43. Springer-Verlag, 1996.

[BHMV94] V. Bruyère, G. Hansel, C. Michaux, and R. Villemaire. Logic and p-recognizable sets of integers. *Bulletin of the Belgian Mathematical Society*, 1(2):191–238, March 1994.

[BJW01] B. Boigelot, S. Jodogne, and P. Wolper. On the use of weak automata for deciding linear arithmetic with integer and real variables. In *IJCAR*, pages 611–625, 2001.

[BL01] B. Boigelot and L. Latour. Counting the solutions of Presburger equations without enumerating them. In *Proc. 6th International Conference on Implementations and Applications of Automata*, number 2494 in Lecture Notes in Computer Science, pages 40–51. Springer-Verlag, 2001.

[Boi99] B. Boigelot. *Symbolic methods for exploring infinite state spaces*. PhD Thesis, Université de Liège, Belgium, 1999.

[Bru85] V. Bruyère. Entiers et automates finis. Mémoire de fin d'études, Université de Mons, Belgium, 1985.

[BRW98] B. Boigelot, S. Rassart, and P. Wolper. On the expressiveness of real and integer arithmetic automata (extended abstract). In *ICALP*, pages 152–163, 1998.

[Buc60] J.R. Buchi. Weak second-order arithmetic and finite automata. *Zeitschrift Math. Logik und Grundlagen der Mathematik*, 6:66–92, 1960.

[BW01] A. Bockmayr and V. Weispfenning. Solving numerical constraints. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, pages 751–842. Elsevier Science, 2001.

[CF89] M. Clausen and A. Fortenbacher. Efficient solution of linear diophantine equations. *Journal of Symbolic Computation*, 8:201–216, 1989.

[CH78] P. Cousot and N. Halbwachs. Automatic discovery of linear restraints among variables of a program. In *Conference Record of the Fifth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 84–97. ACM Press, New York, NY, 1978.

[Clu65] E.J. Cluskey. *Introduction to the theory of switching circuits*. McGrawHill, 1965.

[Cob69] A. Cobham. On the base-dependence of sets of numbers recognizable by finite automata. *Mathematical Systems Theory*, 3:186–192, 1969.

[Dom91] E. Domenjoud. Solving systems of linear diophantine equations: An algebraic approach. In *Mathematical Foundations of Computer Science*, pages 141–150, 1991.

[End01] H. B. Enderton. *A Mathematical Introduction to Logic*. Academic Press, New-York, second edition, 2001.

[FL05] A. Finkel and J. Leroux. The convex hull of a number decision diagram is a computable polyhedron. *To appearn in Information Processing Letters*, 2005.

[Fra94] J.B. Fraleigh. *A First Course in Abstract Algebra*. Addison-Wesley, 1994.

[GP79] F.R. Giles and W.R. Pulleyblank. Total dual integrality and integer polyhedra. *Linear Algebra and its Applications*, 25:191–196, 1979.

[Gra91] P. Granger. Static analysis of linear congruence equalitites among variables of a program. In S. Abramsky and T. S. E. Maibaum, editors, *TAPSOFT'91: Proc. of the International Joint Conference on Theory and Practice of Software Development*, pages 169–192. Springer, Berlin, Heidelberg, 1991.

[GS66] S. Ginsburg and E.H. Spanier. Semigroups, Presburger formulas and languages. *Pacific Journal of Mathematics*, 16(2):285–296, 1966.

[Har65] M.A. Harrison. *Introduction to switching and automata theory*. McGrawHill, 1965.

[Hil90] D. Hilbert. Über die Theorie der algebraische Formen. *Mathematische Annalen*, 36:473–534, 1890.

[Hop71] J.E. Hopcroft. An $n \log n$ algorithm for minimizing states in a finite automaton. *Theory of Machine Computation*, pages 189–196, 1971.

[HU79] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages and Computation*. Addison Wesley, 1979.

[Jac89] N. Jacobson. *Basic algebra, I*. W. H. Freeman and Company, New York, second edition, 1989.

[Kir89] C. Kirchner. From unification in combination of equational theories to a new ac-unification algorithm. In H. Ait-Kaci and M. Nivat, editors, *Resolution of Equations in Algebraic Structures, volume 2*, pages 171–210. Academic Press, New York, 1989.

[Kla04a] F. Klaedtke. *Automata-based Decision Procedures for Weak Arithmetics*. PhD Thesis, Institut für Informatik, Albert-Ludwigs-Universität Freiburg, Freiburg, Germany, 2004.

[Kla04b] F. Klaedtke. On the automata size for Presburger arithmetic. In *Proceedings of the 19th Annual IEEE Symposium on Logic in Computer*

Science (LICS 2004), pages 110–119. IEEE Computer Society Press, 2004.

[LAS] The Liège Automata-based Symbolic Handler (LASH). Available at `http://www.montefiore.ulg.ac.be/~boigelot/research/lash/`.

[Lat04] L. Latour. From automata to formulas: Convex integer polyhedra. In *Proceedings of 19th IEEE Symposium on Logic in Computer Science (LICS 2004)*, pages 120–129. IEEE Computer Society Press, 2004.

[Lat05a] L. Latour. Computing affine hulls over $\mathbb{Q}$ and $\mathbb{Z}$ from sets represented by number decision diagrams. Technical Report 2005-49, Centre Fédéré en Vérification, 2005.

[Lat05b] L. Latour. Computing affine hulls over $\mathbb{Q}$ and $\mathbb{Z}$ from sets represented by number decision diagrams. In *Proceedings of 10th International Conference on Implementations and Applications of Automata*, number 3845 in Lecture Notes in Computer Science, pages 213–224, 2005.

[Ler03] J. Leroux. *Algorithmique de la vérification des systèmes à compteurs. Approximation et accélération. Implémentation de l'outil FAST*. PhD Thesis, Ecole Normale Supérieure de Cachan, Cachan, France, 2003.

[Ler04a] J. Leroux. The affine hull of a binary automaton is computable in polynomial time. *Electr. Notes Theor. Comput. Sci.*, 98:89–104, 2004.

[Ler04b] J. Leroux. A polynomial time Presburger criterion and synthesis for number decision diagram. Technical report, Université de Montréal, 2004.

[Ler05] J. Leroux. A polynomial time presburger criterion and synthesis for number decision diagrams. In *To appear in Proceedings of 20th IEEE Symposium on Logic in Computer Science (LICS 2005)*. IEEE Computer Society Press, 2005.

[Lug04] D. Lugiez. From automata to semi-linear sets: a solution for polyhedra and even more general sets. Technical Report 21-2004, Lab. d'informatique de Marseilles, 2004.

[McN63] R. McNaughton. Review of [Buc60]. *Journal of Symbolic Logic*, 28:100–102, 1963.

[MF71] A.R. Meyer and M.J. Fischer. Economy of description by automata, grammars and formal systems. In *Proc. 12th Annual Symp. on Switching and Automata Theory*, pages 188–191. IEEE Computer Society, 1971.

[MH04] M. Müller-Olm and H. Seidl. Interprocedural analysis of modular arithmetic. Technical Report 789, Fachbereich Informatik, Universiteit Dortmund, 2004.

[Moo71] F.R. Moore. On the bounds for state-set size in the proofs of equivalence between deterministic, non-deterministic, and two-way finite automata. *IEEE Transactions on Computers*, pages 1211–1214, 1971.

[MP86] C. Michaux and F. Point. Les ensembles k-reconnaissables sont définissables dans $\langle \mathbf{N}, +, v_k \rangle$. *Comptes rendus de l'Académie des Sciences de Paris*, 303:939–942, 1986.

[MS04] M. Müller-Olm and H. Seidl. A note on Karr's algorithm. In Josep Diaz, Juhani Karhumäki, and Arto Lepistö, editors, *Proceedings of the 31st International Colloquium on Automata, Languages and Programming (ICALP 2004)*, volume 3142 of *Lecture Notes in Computer Science*, pages 1016–1028. Springer-Verlag Heidelberg, 2004.

[MS05a] M. Müller-Olm and H. Seidl. Analysis of modular arithmetic. In *Proceedings of the European Symposium on Programming (ESOP 2005)*, volume 3444 of *Lecture Notes in Computer Science*, pages 46–60. Springer-Verlag Heidelberg, 2005.

[MS05b] M. Müller-Olm and H. Seidl. A generic framework for interprocedural analysis of numerical properties. In *Proceedings of the 12th International Static Analysis Symposium (SAS 2005)*, volume 3672 of *Lecture Notes in Computer Science*, pages 235–250. Springer-Verlag Heidelberg, 2005.

[Muc03] A. Muchnik. The definable criterion for definability in presburger arithmetic and its applications. *Theoretical Computer Science*, 290(3):1433–1444, 2003.

[MV96] C. Michaux and R. Villemaire. Presburger arithmetic and recognizability of sets of natural numbers by automata: New proofs of Cobham's and Semenov's theorems. *Annals Pure Applied Logic*, 77(3):251–277, 1996.

[NP71] R. Mc Naughton and S. Papert. *Counter-Free Automata*. M.I.T. Press, Cambridge, Mass., 1971.

[OME] The Omega Project: Frameworks and algorithms for the analysis and transformation of scientific programs. Available at `http://www.cs.umd.edu/projects/omega/`.

[Opp78] D. Oppen. A $2^{2^{2^{pn}}}$ upper bound on the complexity of presburger arithmetic. *Journal of Computer and System Sciences*, 16:323–332, July 1978.

[Per90] D. Perrin. Finite automata. In J. Van Leeuwen, editor, *Handbook of Thoretical Computer Science*, pages 1–53. North Holland, 1990.

[Pot91] L. Pottier. Minimal solutions of linear diophantine systems: Bounds and algorithms. In R. V. Book, editor, *Proceedings 4th Conference on Rewriting Techniques and Applications, Como (Italy)*, volume 488, pages 162–173. Springer-Verlag, 1991.

[Pre29] M. Presburger. Über die Volständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. In *Sparawozdanie z I Kongresu matematyków krajów slowianskich*, pages 92–101, 395, Warsaw, Poland, 1929.

[Pre91] M. Presburger. On the completeness of a certain system of arithmetic of whole numbers in which addition occurs as the only operation. *History and Philosophy of Logic*, 12:225–233, 1991. English translation of the article [Pre29] by D. Jacquette.

[RV02] T. Rybina and A. Voronkov. Using canonical representations of solutions to speed up infinite-state model checking. In *Proceedings of the 14th International Conference on Computer Aided Verification*, number 2404 in Lecture Notes In Computer Science, pages 386–400. Springer-Verlag, 2002.

[Sch86] A. Schrijver. *Theory of Linear and Integer Programming*. John Wiley and Sons, New-York, 1986.

[Sem77] A.L. Semenov. Presburgerness of predicates regular in two number systems. *Siberian Mathematical Journal*, 18:289–299, 1977.

[SKR98] T.R. Shiple, J.H. Kukula, and R.K. Ranjan. A comparison of Presburger engines for EFSM reachability. In *Proceedings of the 10th Intl. Conf. on Computer-Aided Verification*, volume 1427 of *Lecture Notes in Computer Science*, pages 280–292, Vancouver, June/July 1998. Springer-Verlag.

[Smi61] H.J.S. Smith. On systems of linear indeterminate equations and congruences. *Philosophical Transactions of the Royal Society of London*, 151:293–326, 1861.

[Sto00] A. Storjohann. *Algorithms for matrix canonical forms*. PhD Thesis, Swiss federal institute, Zurich, 2000.

[Tar72] R.E. Tarjan. Depth first search and linear graph algorithms. *SIAM Journal onComputing*, 1(2):146–160, June 1972.

[vdC31] J.G. van der Corput. Konstruktion der Minimalbasis für spezielle diophantische Systeme von linear-homogenen Gleichungen und Ungleichungen. In *Proceedings Koninlijke Akademie van Wetenschappen te Amsterdam*, volume 34, pages 515–523, 1931.

[Vil92] R. Villemaire. The theory of $\langle \mathbf{N}, +, V_k, V_l \rangle$ is undecidable. *Theoretical Computer Science*, 106:337–349, 1992.

[WB95] P. Wolper and B. Boigelot. An automata-theoretic approach to Presburger arithmetic constraints. In *Proceedings of Static Analysis Symposium*, volume 983 of *Lecture Notes in Computer Science*, pages 21–32, Glasgow, September 1995. Springer-Verlag.

[WB00] P. Wolper and B. Boigelot. On the construction of automata from linear arithmetic constraints. In *Proc. 6th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 1785 of *Lecture Notes in Computer Science*, pages 1–19, Berlin, March 2000. Springer-Verlag.

Index