

# Linear and Integer Optimization (V3C1/F4C1)

Lecture notes

Ulrich Brenner

Research Institute for Discrete Mathematics, University of Bonn

Summer term 2022

July 14, 2022

23:28

# Preface

Continuous updates of these lecture notes can be found on the webpage of the lecture course: [http://www.or.uni-bonn.de/lectures/ss22/lgo\\_ss22.html](http://www.or.uni-bonn.de/lectures/ss22/lgo_ss22.html)

These lecture notes are based on a number of textbooks and lecture notes from earlier courses. See e.g. the lecture notes by Tim Nieberg (winter term 2012/2013) and Stephan Held (winter term 2013/2014 and 2017/18) that are available online on the teaching web pages of the Research Institute for Discrete Mathematics, University of Bonn (<http://www.or.uni-bonn.de/lectures>).

## Recommended textbooks:

- Chvátal [1983]: Still a good introduction into the field of linear programming.
- Korte and Vygen [2018]: Chapters 3–5 contain the most important results of this lecture course. Very compact description.
- Matoušek and Gärtner [2007]: Very good description of the linear programming part. For some results, proofs are missing, and the book does not consider integer programming.
- Schrijver [1986]: Comprehensive textbook covering both linear and integer programming. Proofs are short but precise.

Prerequisites of this course are the lectures “Algorithmische Mathematik I” and “Lineare Algebra I/II”. The lecture “Algorithmische Mathematik I” is covered by the textbook by Hougardy and Vygen [2018]. The results concerning Linear Algebra that are used in this course can be found, e.g., in the textbooks by Anthony and Harvey [2012], Bosch [2007], and Fischer [2009].

We we also make use of some basic results of the complexity theory as they are taught in the lecture course “Einführung in die Diskrete Mathematik”. These results on complexity theory can be found e.g. in Chapter 15 of the textbook by Korte and Vygen [2018].

The notation concerning graphs is based on the notation proposed in the textbook by Korte and Vygen [2018].

Please report any errors in these lecture notes to [brenner@or.uni-bonn.de](mailto:brenner@or.uni-bonn.de)

# 1 Introduction

## 1.1 A First Example

Assume that a farmer has 10 hectares of land where he can grow two kinds of crops: corn and wheat (or a combination of both). For each hectare of corn he gets a revenue of 2 units of money and for each hectare of wheat he gets 3 units of money. Planting corn in an area of one hectare takes him 1 day while planting wheat takes him 2 days per hectare. In total, he has 16 days for the work on his field. Moreover, each hectare planted with corn needs 5 units of water and each hectare planted with wheat needs 2 units of water. In total he has 40 units of water. How can he maximize his revenue?

If  $x_1$  is the number of hectares planted with corn and  $x_2$  is the number of hectares planted with wheat we can write the corresponding optimization problem in the following compact way:

$$\begin{array}{llll} \max & 2x_1 + 3x_2 & & // \text{ Objective function} \\ \text{subject to} & x_1 + x_2 \leq 10 & & // \text{ Bound on the area} \\ & x_1 + 2x_2 \leq 16 & & // \text{ Bound on the workload} \\ & 5x_1 + 2x_2 \leq 40 & & // \text{ Bound on the water resources} \\ & x_1, x_2 \geq 0 & & // \text{ An area cannot be negative} \end{array}$$

This is what we call a linear program (LP). In such an LP, we are given a linear objective function (in our case  $(x_1, x_2) \mapsto 2x_1 + 3x_2$ ) that has to be maximized or minimized under a number of linear constraints. These constraints can be given by linear inequalities (but not strict inequalities “ $<$ ”) or by linear equations. However, a linear equation can easily be replaced by a pair of inequalities (e.g.  $4x_1 + 3x_2 = 7$  is equivalent to  $4x_1 + 3x_2 \leq 7$  and  $4x_1 + 3x_2 \geq 7$ ), so we may assume that all constraints are given by linear inequalities.

In our example, there were only two variables,  $x_1$  and  $x_2$ . In this case, linear programs can be solved graphically. Figure 1 illustrates the method. The grey area is the set

$$\{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 + x_2 \leq 10\} \cap \{(x_1, x_2) \in \mathbb{R}^2 \mid x_1 + 2x_2 \leq 16\} \cap \{(x_1, x_2) \in \mathbb{R}^2 \mid 5x_1 + 2x_2 \leq 40\} \cap \{(x_1, x_2) \in \mathbb{R}^2 \mid x_1, x_2 \geq 0\},$$

which is the set of all feasible solutions of our problem. We can solve the problem by moving the green line, which is orthogonal to the cost vector  $\begin{pmatrix} 2 \\ 3 \end{pmatrix}$  (shown in red), in the direction of  $\begin{pmatrix} 2 \\ 3 \end{pmatrix}$  as long as it intersects the feasible area. We end up with  $x_1 = 4$  and  $x_2 = 6$ , which is in fact an optimum solution.

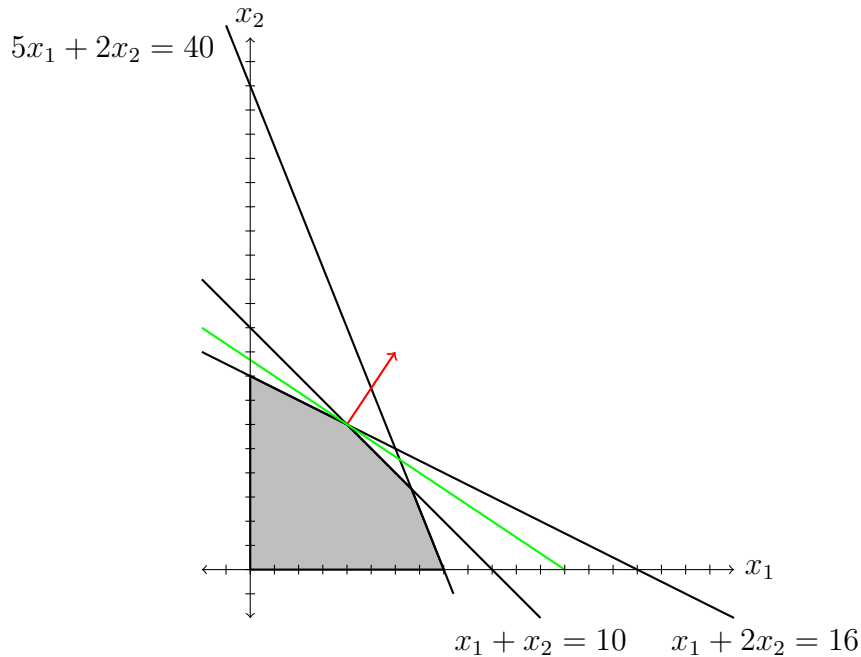


Fig. 1: Graphic solution of the first example.

## 1.2 Optimization Problems

**Definition 1** An **optimization problem** is a pair  $(I, f)$  where  $I$  is a set and  $f : I \rightarrow \mathbb{R}$  is a mapping. The elements of  $I$  are called **feasible solutions** of  $(I, f)$ . If  $I = \emptyset$ , the optimization problem  $(I, f)$  is called **infeasible**, otherwise we call it **feasible**. The function  $f$  is called the **objective function** of  $(I, f)$ .

We either ask for an element  $x^* \in I$  such that for all  $x \in I$  we have  $f(x) \leq f(x^*)$  (then  $(I, f)$  is called a **maximization problem**) or for an element  $x^* \in I$  such that for all  $x \in I$  we have  $f(x) \geq f(x^*)$  (then  $(I, f)$  is called a **minimization problem**). In both cases, such an element  $x^*$  is called an **optimum solution**.  $(I, f)$  is **unbounded** if for all  $K \in \mathbb{R}$ , there is an  $x \in I$  with  $f(x) > K$  (for the maximization problem) or an  $x \in I$  with  $f(x) < K$  (for the minimization problem). An optimization problem is called **bounded** if it is not unbounded.

In this lecture course, we consider optimization problems with linear objective functions and linear constraints. The constraints can be written in a compact way using matrices:

**LINEAR PROGRAMMING**

*Instance:* A matrix  $A \in \mathbb{R}^{m \times n}$ , vectors  $c \in \mathbb{R}^n$  and  $b \in \mathbb{R}^m$ .

*Task:* Find a vector  $x \in \mathbb{R}^n$  with  $Ax \leq b$  maximizing  $c^t x$ .

**Notation:** Unless stated differently, always let  $A = (a_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}} \in \mathbb{R}^{m \times n}$ ,  $b = (b_1, \dots, b_m) \in \mathbb{R}^m$  and  $c = (c_1, \dots, c_n) \in \mathbb{R}^n$ .

**Remark:** Real vectors are simply ordered sets of real numbers. But when we multiply vectors with each other or with matrices, we have to interpret them as  $n \times 1$ -matrices (column vectors) or as  $1 \times n$ -matrices (row vectors). By default, we consider vectors as column vectors in this context, so if we want to use them as row vectors, we have to transpose them (“ $c^t$ ”).

We often write linear programs in the following way:

$$\begin{aligned} & \max c^t x \\ \text{s.t.} \quad & Ax \leq b \end{aligned} \tag{1}$$

Or, in a shorter version we write:  $\max\{c^t x \mid Ax \leq b\}$ .

The  $i$ -th row of matrix  $A$  encodes the constraint  $\sum_{j=1}^n a_{ij}x_j \leq b_i$  on a solution  $x = (x_1, \dots, x_n)$ . We could also allow equation constraints of the form  $\sum_{j=1}^n a_{ij}x_j = b_i$  but (as mentioned in the example in Section 1.1) these could be easily replaced by two inequalities. The formulation (1) which avoids such equation constraints is called **standard inequality form**. Obviously, we can also handle minimization problems with this approach because minimizing the objective function  $c^t x$  means maximizing the objective function  $-c^t x$ .

A second important standard form for linear programs is the **standard equation form**:

$$\begin{aligned} & \max c^t x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned} \tag{2}$$

Both standard forms can be transformed into each other: If we are given a linear program in standard equation form we can replace each equation by a pair of inequalities and the constraint  $x \geq 0$  by  $-I_n x \leq 0$  (where  $I_n$  is always the  $n \times n$ -identity matrix). This leads to a formulation of the same linear program in standard inequality form.

The transformation from the standard inequality form into the standard equation form is slightly more complicated: Assume we are given the following linear program in standard inequality form

$$\begin{aligned} & \max c^t x \\ \text{s.t.} \quad & Ax \leq b \end{aligned} \tag{3}$$

We replace each variable  $x_i$  by two variables  $z_i$  and  $\bar{z}_i$ . Moreover, for each of the  $m$  constraints we introduce a new variable  $\tilde{x}_i$  (a so-called **slack variable**). With variables  $z = (z_1, \dots, z_n)$ ,  $\bar{z} = (\bar{z}_1, \dots, \bar{z}_n)$  and  $\tilde{x} = (\tilde{x}_1, \dots, \tilde{x}_m)$ , we state the following LP in standard equation form:

$$\begin{aligned} & \max c^t(z - \bar{z}) \\ \text{s.t.} \quad & [A \mid -A \mid I_m] \begin{pmatrix} z \\ \bar{z} \\ \tilde{x} \end{pmatrix} = b \\ & z, \bar{z}, \tilde{x} \geq 0 \end{aligned} \tag{4}$$

Note that  $[A \mid -A \mid I_m]$  is the  $m \times 2n + m$ -matrix that we get by concatenating the matrices  $A$ ,  $-A$  and  $I_m$ . Any solution  $z, \bar{z}$  and  $\tilde{x}$  of the LP (4) gives a solution of the LP (3) with the same cost by setting:  $x_j := z_j - \bar{z}_j$  (for  $j \in \{1, \dots, n\}$ ).

On the other hand, if  $x$  is a solution of LP (3), then we get a solution of LP (4) with the same cost by setting  $z_j := \max\{x_j, 0\}$ ,  $\bar{z}_j := -\min\{x_j, 0\}$  (for  $j \in \{1, \dots, n\}$ ) and  $\tilde{x}_i = b_i - \sum_{j=1}^n a_{ij}x_j$  (for  $i \in \{1, \dots, m\}$ , where  $\sum_{j=1}^n a_{ij}x_j \leq b_i$  is the  $i$ -th constraint of  $Ax \leq b$ ).

Note that (in contrast to the first transformation) this second transformation (from the standard inequality form into the standard equation form) leads to a different solution space because we have to introduce new variables.

### 1.3 Possible Outcomes

There are three possible outcomes for a linear program  $\max\{c^t x \mid Ax \leq b\}$ :

- The linear program can be **infeasible**. This means that  $\{x \in \mathbb{R}^n \mid Ax \leq b\} = \emptyset$ . A simple example is:

$$\begin{aligned} & \max x \\ \text{s.t.} \quad & x \leq 0 \\ & -x \leq -1 \end{aligned} \tag{5}$$

- The linear program can be **feasible but unbounded**. This means that for each constant  $K$  there is a feasible solution  $x$  with  $c^t x \geq K$ . An example is

$$\begin{aligned} & \max x \\ \text{s.t.} \quad & x - y \leq 0 \\ & y - x \leq 1 \end{aligned} \tag{6}$$

- The linear program can be **feasible and bounded**, so there is an  $x \in \mathbb{R}^n$  with  $Ax \leq b$  and we have  $\sup\{c^t x \mid Ax \leq b\} < \infty$ . An example is the LP that we saw in Section 1.1. It will turn out that in this case there is always a vector  $\tilde{x} \in \mathbb{R}^n$  with  $A\tilde{x} \leq b$  with  $c^t \tilde{x} = \sup\{c^t x \mid Ax \leq b\}$ .

We will see that deciding if a linear program is feasible is as hard as computing an optimum solution to a feasible and bounded linear program (see Section 2.4).

### 1.4 Integrality Constraints

In many applications, we need an integral solution. This leads to the following class of problems:

<p>INTEGER LINEAR PROGRAMMING</p> <p><i>Instance:</i> A matrix <math>A \in \mathbb{R}^{m \times n}</math>, vectors <math>c \in \mathbb{R}^n</math> and <math>b \in \mathbb{R}^m</math>.</p> <p><i>Task:</i> Find a vector <math>x \in \mathbb{Z}^n</math> with <math>Ax \leq b</math> maximizing <math>c^t x</math>.</p>
--

Replacing the constraint  $x \in \mathbb{R}^n$  by  $x \in \mathbb{Z}^n$  makes a huge difference. We will see that there are polynomial-time algorithms for LINEAR PROGRAMMING while INTEGER LINEAR PROGRAMMING is NP-hard.

Of course, one can also consider optimization problems where we have integrality constraints only for some of the variables. These linear optimization problems are called MIXED INTEGER LINEAR PROGRAMS.

## 1.5 Modeling of Optimization Problems as (Integral) Linear Programs

We consider some examples how optimization problems can be modeled as LPs or ILPs. Many flow problems can easily be formulated as linear programs:

**Definition 2** Let  $G$  be a directed graph with capacities  $u : E(G) \rightarrow \mathbb{R}_{>0}$  and let  $s$  and  $t$  be two vertices of  $G$ . A feasible  $s$ - $t$ -flow in  $(G, u)$  is a mapping  $f : E(G) \rightarrow \mathbb{R}_{\geq 0}$  with

- $f(e) \leq u(e)$  for all  $e \in E(G)$  and
- $\sum_{e \in \delta_G^+(v)} f(e) - \sum_{e \in \delta_G^-(v)} f(e) = 0$  for all  $v \in V(G) \setminus \{s, t\}$ .

The **value** of an  $s$ - $t$ -flow  $f$  is  $\text{val}(f) = \sum_{e \in \delta_G^+(s)} f(e) - \sum_{e \in \delta_G^-(s)} f(e)$ .

### MAXIMUM-FLOW PROBLEM

*Instance:* A directed Graph  $G$ , capacities  $u : E(G) \rightarrow \mathbb{R}_{>0}$ , vertices  $s, t \in V(G)$  with  $s \neq t$ .

*Task:* Find an  $s$ - $t$ -flow  $f : E(G) \rightarrow \mathbb{R}_{\geq 0}$  of maximum value.

This problem can be formulated as a linear program in the following way:

$$\begin{aligned}
 \max \quad & \sum_{e \in \delta_G^+(s)} x_e - \sum_{e \in \delta_G^-(s)} x_e \\
 \text{s.t.} \quad & x_e \geq 0 \quad \text{for } e \in E(G) \\
 & x_e \leq u(e) \quad \text{for } e \in E(G) \\
 & \sum_{e \in \delta_G^+(v)} x_e - \sum_{e \in \delta_G^-(v)} x_e = 0 \quad \text{for } v \in V(G) \setminus \{s, t\}
 \end{aligned} \tag{7}$$

It is well known that the value of a maximum  $s$ - $t$ -flow equals the capacity of a minimum cut separating  $s$  from  $t$ . We will see in Section 2.5 that this result also follows from properties of the linear program formulation. Moreover, if the capacities are integral, there is always a maximum flow that is integral (see Section 8.4).

In some cases, we first have to modify a given optimization problem slightly in order to get a linear program formulation. See the following example of a modified version of the MAXIMUM-FLOW PROBLEM where we have two sources and want to maximize the minimal out-flow of both sources.

**BOTTLENECK MAXIMUM-FLOW PROBLEM WITH 2 SOURCES**

*Instance:* A directed Graph  $G$ , capacities  $u : E(G) \rightarrow \mathbb{R}_{>0}$ , three vertices  $s_1, s_2, t \in V(G)$ .

*Task:* Find a mapping  $f : E(G) \rightarrow \mathbb{R}_{\geq 0}$  with

- $f(e) \leq u(e)$  for all  $e \in E(G)$  and
- $\Delta_f(v) = 0$  for all  $v \in V(G) \setminus \{s_1, s_2, t\}$

such that  $\min\{\Delta_f(s_1), \Delta_f(s_2)\}$  is maximized

The objective function here is not a linear function but the minimum of two linear function. To see how such a problem can be written as an LP, we assume slightly more general that we are given the following optimization problem:

$$\begin{array}{ll} \max & \min\{c^t x + d, e^t x + f\} \\ \text{s.t.} & Ax \leq b \end{array}$$

for some  $c, e \in \mathbb{R}^n$  and  $d, f \in \mathbb{R}$ .

This is not a linear program because the objective function is not a linear function but a piecewise linear function. Though the objective function is not linear, we can define an equivalent linear program in the following way:

$$\begin{array}{ll} \max & \sigma \\ \text{s.t.} & \sigma - c^t x \leq d \\ & \sigma - e^t x \leq f \\ & Ax \leq b \end{array}$$

And of course, this trick also works if we want to compute the minimum of more than two linear functions.

More or less the same trick can be applied to the following problem in which the objective function contains absolute values of linear functions:

$$\begin{array}{ll} \min & |c^t x + d| \\ \text{s.t.} & Ax \leq b \end{array}$$

for some  $c \in \mathbb{R}^n$  and  $d \in \mathbb{R}$ . Again the problem can be written equivalently as a linear program



in the following form:

$$\begin{aligned} \max \quad & -\sigma \\ \text{s.t.} \quad & -\sigma - c^t x \leq d \\ & -\sigma + c^t x \leq -d \\ & Ax \leq b \end{aligned}$$

The two additional constraints on  $\sigma$  ensure that we have  $\sigma \geq \max\{c^t x + d, -c^t x - d\} = |c^t x + d|$ .

Other problems allow a formulation as an ILP but assumably not an LP formulation:

<p><b>VERTEX COVER PROBLEM</b></p> <p><i>Instance:</i> An undirected graph <math>G</math>, weights <math>c : V(G) \rightarrow \mathbb{R}_{\geq 0}</math>.</p> <p><i>Task:</i> Find a set <math>X \subseteq V(G)</math> with <math>\{v, w\} \cap X \neq \emptyset</math> for all <math>e = \{v, w\} \in E(G)</math> such that <math>\sum_{v \in X} c(v)</math> is minimized.</p>	
---	--

This problem is known to be NP-hard (see standard textbooks like Korte and Vygen [2018]), so we cannot hope for a polynomial-time algorithm. Nevertheless, the problem can easily be formulated as an integer linear program:

$$\begin{aligned} \min \quad & \sum_{v \in V(G)} x_v c(v) \\ \text{s.t.} \quad & x_v + x_w \geq 1 && \text{for } \{v, w\} \in E(G) \\ & x_v \in \{0, 1\} && \text{for } v \in V(G) \end{aligned} \tag{8}$$

For each vertex  $v \in V(G)$ , we have a 0-1-variable  $x_v$  which is 1 if and only if  $v$  should be in the set  $X$ , i.e. if  $(x_v)_{v \in V(G)}$  is an optimum solution to (8), the set  $X = \{v \in V(G) \mid x_v = 1\}$  is an optimum solution to the VERTEX COVER PROBLEM.

This example shows that INTEGER LINEAR PROGRAMMING itself is an NP-hard problem. By skipping the integrality constraints ( $x_v \in \{0, 1\}$ ) we get the following linear program:

$$\begin{aligned} \min \quad & \sum_{v \in V(G)} x_v c(v) \\ \text{s.t.} \quad & x_v + x_w \geq 1 && \text{for } \{v, w\} \in E(G) \\ & x_v \geq 0 && \text{for } v \in V(G) \\ & x_v \leq 1 && \text{for } v \in V(G) \end{aligned} \tag{9}$$

We call this linear program an **LP-relaxation** of (8). In this particular case, the relaxation gives a 2-approximation of the VERTEX COVER PROBLEM: For any solution  $x$  of the relaxed problem, we get an integral solution  $\tilde{x}$  by setting

$$\tilde{x}_v = \begin{cases} 1 & : x_v \geq \frac{1}{2} \\ 0 & : x_v < \frac{1}{2} \end{cases}$$

It is easy to check that yields a feasible solution of the ILP with  $\sum_{v \in V(G)} \tilde{x}_v c(v) \leq 2 \sum_{v \in V(G)} x_v c(v)$ .

Obviously, in minimization problems relaxing some constraints can only decrease the value of an optimum solution. We call the supremum of the ratio between the values of the optimum

solutions of an ILP and its LP-relaxation the **integrality gap** of the relaxation. The rounding procedure described above also proves that in this case the integrality gap is at most 2. Indeed, 2 is the integrality gap as the example of a complete graph with weights  $c(v) = 1$  for all vertices  $v$  shows. For the MAXIMUM-FLOW PROBLEM with integral edge capacities, the integrality gap is 1 because there is always an optimum flow that is integral.

The following problem is NP-hard as well:

STABLE SET PROBLEM

*Instance:* An undirected graph  $G$ , weights  $c : V(G) \rightarrow \mathbb{R}_{\geq 0}$ .

*Task:* Find a set  $X \subseteq V(G)$  with  $|\{v, w\} \cap X| \leq 1$  for all  $e = \{v, w\} \in E(G)$  such that  $\sum_{v \in X} c(v)$  is maximized.

Again, this problem can easily be formulated as an integer linear program:

$$\begin{aligned}
 & \max \sum_{v \in V(G)} x_v c(v) \\
 \text{s.t.} \quad & x_v + x_w \leq 1 && \text{for } \{v, w\} \in E(G) \\
 & x_v \in \{0, 1\} && \text{for } v \in V(G)
 \end{aligned} \tag{10}$$

An LP-relaxation looks like this:

$$\begin{aligned}
 & \max \sum_{v \in V(G)} x_v c(v) \\
 \text{s.t.} \quad & x_v + x_w \leq 1 && \text{for } \{v, w\} \in E(G) \\
 & x_v \geq 0 && \text{for } v \in V(G) \\
 & x_v \leq 1 && \text{for } v \in V(G)
 \end{aligned} \tag{11}$$

Unfortunately, in this case, the LP-relaxation is of no use. Even if  $G$  is a complete graph (were a feasible solution of the STABLE SET PROBLEM can contain at most one vertex), setting  $x_v = \frac{1}{2}$  for all  $v \in V(G)$  would be a feasible solution of the LP-relaxation. This example shows that the integrality gap is at least  $\frac{n}{2}$ . Hence, this LP-relaxation does not provide any useful information about a good ILP solution.

## 1.6 Polyhedra

In this section, we examine basic properties of solution spaces of linear programs.

**Definition 3** Let  $X \subseteq \mathbb{R}^n$  (for  $n \in \mathbb{N}$ ).  $X$  is called **convex** if for all  $x, y \in X$  and  $t \in [0, 1]$  we have  $tx + (1 - t)y \in X$ .

**Definition 4** For  $x_1, \dots, x_k \in \mathbb{R}^n$ ,  $\lambda_1, \dots, \lambda_k$ ,  $\lambda_i \geq 0$  ( $i \in \{1, \dots, k\}$ ) with  $\sum_{i=1}^k \lambda_i = 1$ , we call  $x = \sum_{i=1}^k \lambda_i x_i$  **convex combination** of  $x_1, \dots, x_k$ . The **convex hull**  $\text{conv}(X)$  of a set  $X \subseteq \mathbb{R}^n$  is the set of all convex combinations of sets of vectors in  $X$ .

**Remark:** It is easy to check that the convex hull of a set  $X \subseteq \mathbb{R}^n$  is the (inclusion-wise) minimal convex set containing  $X$ .

**Definition 5** Let  $X \subseteq \mathbb{R}^n$  for some  $n \in \mathbb{N}$ .

- (a)  $X$  is called a **half-space** if there is a vector  $a \in \mathbb{R}^n \setminus \{0\}$  and a number  $b \in \mathbb{R}$  such that  $X = \{x \in \mathbb{R}^n \mid a^t x \leq b\}$ . The vector  $a$  is called a **normal** of  $X$ .
- (b)  $X$  is called a **hyperplane** if there is a vector  $a \in \mathbb{R}^n \setminus \{0\}$  and a number  $b \in \mathbb{R}$  such that  $X = \{x \in \mathbb{R}^n \mid a^t x = b\}$ . The vector  $a$  is called a **normal** of  $X$ .
- (c)  $X$  is called a **polyhedron** if there are a matrix  $A \in \mathbb{R}^{m \times n}$  and a vector  $b \in \mathbb{R}^m$  such that  $X = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ .
- (d)  $X$  is called a **polytope** if it is a polyhedron and there is a number  $K \in \mathbb{R}$  such that  $\|x\| \leq K$  for all  $x \in X$ .

**Examples:** The empty set is a polyhedron because  $\emptyset = \{x \in \mathbb{R}^n \mid 0^t x \leq -1\}$  and, of course, it is a polytope.  $\mathbb{R}^n$  is also a polyhedron, because  $\mathbb{R}^n = \{x \in \mathbb{R}^n \mid 0^t x \leq 0\}$  (but, of course,  $\mathbb{R}^n$  is not a polytope).

**Observation:** Polyhedra are convex and closed (see exercises).

**Lemma 1** A set  $X \subseteq \mathbb{R}^n$  is a polyhedron if and only if one of the following conditions holds:

- $X = \mathbb{R}^n$
- $X$  is the intersection of a finite number of half-spaces.

**Proof:** “ $\Leftarrow$ ” If  $X = \mathbb{R}^n$  or  $X$  is the intersection of a finite number of half-spaces, it is obviously a polyhedron.

“ $\Rightarrow$ ” Assume that  $X$  is a polyhedron but  $X \neq \mathbb{R}^n$ . If  $X = \emptyset$ , then  $X = \{x \in \mathbb{R}^n \mid \mathbb{1}_n^t x \leq -1\} \cap \{x \in \mathbb{R}^n \mid -\mathbb{1}_n^t x \leq -1\}$  (where  $\mathbb{1}_n$  is the all-one vector of length  $n$ ). Hence we can assume that  $X \neq \emptyset$ .

Let  $A \in \mathbb{R}^{m \times n}$  be a matrix and  $b \in \mathbb{R}^m$  a vector with  $X = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . Denote the rows

of  $A$  by  $a_1, \dots, a_m$ . If  $a_j = 0$  for an  $j \in \{1, \dots, m\}$ , then  $b_j \geq 0$  (where  $b = (b_1, \dots, b_m)$ ) because otherwise  $X = \emptyset$ . Hence we have

$$X = \bigcap_{j=1}^m \{x \in \mathbb{R}^n \mid a_j^t x \leq b_j\} = \bigcap_{j=1, \dots, m: a_j \neq 0} \{x \in \mathbb{R}^n \mid a_j^t x \leq b_j\},$$

which is a representation of  $X$  as an intersection of a finite number of half-spaces.  $\square$

**Definition 6** *The dimension of a set  $X \subseteq \mathbb{R}^n$  is*

$$\dim(X) = n - \max\{\text{rank}(A) \mid A \in \mathbb{R}^{n \times n} \text{ with } Ax = Ay \text{ for all } x, y \in X\}.$$

In other words, the dimension of  $X \subseteq \mathbb{R}^n$  is  $n$  minus the maximum size of a set of linear independent vectors that are orthogonal to any difference of elements in  $X$ . For example, the empty set and sets consisting of exactly one vector have dimension 0. The set  $\mathbb{R}^n$  has dimension  $n$ .

**Observation:** The dimension of a set  $X \subseteq \mathbb{R}^n$  is the largest  $d$  for which  $X$  contains elements  $v_0, v_1, \dots, v_d$  such that  $v_1 - v_0, v_2 - v_0, \dots, v_d - v_0$  are linearly independent.

**Definition 7** *A set  $X \subseteq \mathbb{R}^n$  is called a **convex cone** if  $X \neq \emptyset$  and for all  $x, y \in X$  and  $\lambda, \mu \in \mathbb{R}_{\geq 0}$  we have  $\lambda x + \mu y \in X$ .*

**Observation:** A non-empty set  $X \subseteq \mathbb{R}^n$  is a convex cone if and only if  $X$  is convex and for all  $x \in X$  and  $\lambda \in \mathbb{R}_{\geq 0}$  we have  $\lambda x \in X$ .

**Definition 8** *A set  $X \subseteq \mathbb{R}^n$  is called a **polyhedral cone** if it is a polyhedron and a convex cone.*

**Lemma 2** *A set  $X \subseteq \mathbb{R}^n$  is a polyhedral cone if and only if there is a matrix  $A \in \mathbb{R}^{m \times n}$  such that  $X = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$ .*

**Proof:** “ $\Leftarrow$ ” Let  $X = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$  for some matrix  $A \in \mathbb{R}^{m \times n}$ . Then  $X$  obviously is a polyhedron and non-empty (because  $0 \in X$ ). And if  $x, y \in X$  and  $\lambda, \mu \in \mathbb{R}_{\geq 0}$ , then  $A(\lambda x + \mu y) \leq 0$ , so  $\lambda x + \mu y \in X$ . Hence  $X$  is a convex cone, too.

“ $\Rightarrow$ ” Let  $X \subseteq \mathbb{R}^n$  be a polyhedral cone. In particular, there is a matrix  $A \in \mathbb{R}^{m \times n}$  and a vector  $b \in \mathbb{R}^m$  such that  $X = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . Since  $X$  is a convex cone, it is non-empty and it must contain 0. Therefore, no entry of  $b$  can be negative. Thus,  $X \supseteq \{x \in \mathbb{R}^n \mid Ax \leq 0\}$ . But if there was a vector  $x \in X$  such that  $Ax$  has positive  $i$ -th entry (for an  $i \in \{1, \dots, m\}$ ), then for

sufficiently large  $\lambda$ , the  $i$ -th entry of  $\lambda Ax$  would be greater than  $b_i$  which is a contradiction to the assumption that  $X$  is a convex cone. Therefore,  $X = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$ .  $\square$

Let  $x_1, \dots, x_m \in \mathbb{R}^n$  be vectors. The cone **generated** by  $x_1, \dots, x_m$  is the set

$$\text{cone}(\{x_1, \dots, x_m\}) := \left\{ \sum_{i=1}^m \lambda_i x_i \mid \lambda_1, \dots, \lambda_m \geq 0 \right\}.$$

A convex cone  $C$  is called **finitely generated** if there are vectors  $x_1, \dots, x_m \in \mathbb{R}^n$  with  $C = \text{cone}(\{x_1, \dots, x_m\})$ .

It is easy to check that  $\text{cone}(\{x_1, \dots, x_m\})$  is indeed a convex cone. We will see in Section 3.5 that a cone is polyhedral if and only if it is finitely generated.



## 2 Duality

### 2.1 Dual LPs

Consider the following linear program (P):

$$\begin{array}{ll} \max & 12x_1 + 10x_2 \\ \text{s.t.} & 4x_1 + 2x_2 \leq 5 \\ & 8x_1 + 12x_2 \leq 7 \\ & 2x_1 - 3x_2 \leq 1 \end{array}$$

How can we find upper bounds on the value of an optimum solution? By combining the first two constraints we can get the following bound for any feasible solution  $(x_1, y_1)$ :

$$12x_1 + 10x_2 = 2 \cdot (4x_1 + 2x_2) + \frac{1}{2}(8x_1 + 12x_2) \leq 2 \cdot 5 + \frac{1}{2} \cdot 7 = 13.5.$$

We can even do better by combining the last two inequalities:

$$12x_1 + 10x_2 = \frac{7}{6} \cdot (8x_1 + 12x_2) + \frac{4}{3} \cdot (2x_1 - 3x_2) \leq \frac{7}{6} \cdot 7 + \frac{4}{3} \cdot 1 = 9.5.$$

More generally, for computing upper bounds we ask for non-negative numbers  $u_1, u_2, u_3$  such that

$$12x_1 + 10x_2 = u_1 \cdot (4x_1 + 2x_2) + u_2 \cdot (8x_1 + 12x_2) + u_3 \cdot (2x_1 - 3x_2).$$

Then,  $5 \cdot u_1 + 7 \cdot u_2 + 1 \cdot u_3$  is an upper bound on the value of any solution of (P), so we want to chose  $u_1, u_2, u_3$  in such a way that  $5 \cdot u_1 + 7 \cdot u_2 + 1 \cdot u_3$  is minimized.

This leads us to the following linear program (D):

$$\begin{array}{llll} \min & 5u_1 & + & 7u_2 & + & u_3 \\ \text{s.t.} & 4u_1 & + & 8u_2 & + & 2u_3 & = & 12 \\ & 2u_1 & + & 12u_2 & - & 3u_3 & = & 10 \\ & u_1 & & & & & \geq & 0 \\ & & & u_2 & & & \geq & 0 \\ & & & & & u_3 & \geq & 0 \end{array}$$

This linear program is called the **dual linear program of (P)**. Any solution of (D) yields an upper bound on the optimum value of of (P), and in this particular case it turns out that  $u_1 = 0$ ,  $u_2 = \frac{7}{6}$ ,  $u_3 = \frac{4}{3}$  (the second solution from above) with value 9.5 is an optimum solution of (D) because  $x_1 = \frac{11}{16}$ ,  $x_2 = \frac{1}{8}$  is a solution of (P) with value 9.5.

For a general linear program (P)

$$\begin{array}{ll} \max & c^t x \\ \text{s.t.} & Ax \leq b \end{array}$$

in standard inequality form we define its dual linear program (D) as

$$\begin{aligned} & \min b^t y \\ \text{s.t. } & A^t y = c \\ & y \geq 0 \end{aligned}$$

In this context, we call the linear program (P) **primal linear program**.

**Remark:** Note that the dual linear program does not only depend on the objective function and the solution space of the primal linear program but on its description by linear inequalities. For example adding redundant inequalities to the system  $Ax \leq b$  will lead to more variables in the dual linear program.

**Proposition 3** (*Weak duality*) *If both the equation systems  $Ax \leq b$  and  $A^t y = c, y \geq 0$  have a feasible solution, then*

$$\max\{c^t x \mid Ax \leq b\} \leq \min\{b^t y \mid A^t y = c, y \geq 0\}.$$

**Proof:** For  $x$  with  $Ax \leq b$  and  $y$  with  $A^t y = c, y \geq 0$ , we have

$$c^t x = (A^t y)^t x = y^t Ax \leq y^t b.$$

□

**Remark:** The term “dual” implies that applying the transformation from (P) to (D) twice yields (P) again. This is not exactly the case but it is not very difficult to see that dualizing (D) (after transforming it into standard equational form) gives a linear program that is equivalent to (P) (see the exercises).

## 2.2 Fourier-Motzkin Elimination

Consider the following system of inequalities:

$$\begin{aligned} 3x + 2y + 4z &\leq 10 \\ 3x &+ 2z \leq 9 \\ 2x - y &\leq 5 \\ -x + 2y - z &\leq 3 \\ -2x &\leq 4 \\ &2y + 2z \leq 7 \end{aligned} \tag{12}$$

Assume that we just want to decide if a feasible solution  $x, y, z$  exists. The goal is to get rid of the variables one after the other. To get rid of  $x$ , we first reformulate the inequalities such that



we can easily see lower and upper bounds for  $x$ :

$$\begin{aligned}
x &\leq \frac{10}{3} - \frac{2}{3}y - \frac{4}{3}z \\
x &\leq 3 - \frac{2}{3}z \\
x &\leq \frac{5}{2} + \frac{1}{2}y \\
x &\geq -3 + 2y - z \\
x &\geq -2 \\
2y + 2z &\leq 7
\end{aligned} \tag{13}$$

This system of inequalities has a feasible solution if and only if the following system (that does not contain  $x$ ) has a solution:

$$\min \left\{ \frac{10}{3} - \frac{2}{3}y - \frac{4}{3}z, \quad 3 - \frac{2}{3}z, \quad \frac{5}{2} + \frac{1}{2}y \right\} \geq \max \left\{ -3 + 2y - z, \quad -2 \right\} \\
2y + 2z \leq 7 \tag{14}$$

This system can be rewritten equivalently in the following way:

$$\begin{aligned}
\frac{10}{3} - \frac{2}{3}y - \frac{4}{3}z &\geq -3 + 2y - z \\
\frac{10}{3} - \frac{2}{3}y - \frac{4}{3}z &\geq -2 \\
3 - \frac{2}{3}z &\geq -3 + 2y - z \\
3 - \frac{2}{3}z &\geq -2 \\
\frac{5}{2} + \frac{1}{2}y &\geq -3 + 2y - z \\
\frac{5}{2} + \frac{1}{2}y &\geq -2 \\
2y + 2z &\leq 7
\end{aligned} \tag{15}$$

This is equivalent to the following system in standard form:

$$\begin{aligned}
\frac{8}{3}y + \frac{1}{3}z &\leq \frac{19}{3} \\
\frac{2}{3}y + \frac{4}{3}z &\leq \frac{16}{3} \\
2y - \frac{1}{3}z &\leq 6 \\
\frac{2}{3}z &\leq 5 \\
\frac{3}{2}y - z &\leq \frac{11}{2} \\
-\frac{1}{2}y &\leq \frac{9}{2} \\
2y + 2z &\leq 7
\end{aligned} \tag{16}$$

We can iterate this step until we end up with a system of inequalities without variables. It is easy to check if all inequalities in this final system are valid, which is equivalent to the existence of a feasible solution of the initial system of inequalities. Moreover, we can also find a feasible solution if one exists. To see this, note that any solution of the system (14) (that contains  $y$  and  $z$  as variables only) also gives a solution of the system (13) by setting  $x$  to a value in the interval

$$\left[ \max \left\{ -3 + 2y - z, \quad -2 \right\}, \min \left\{ \frac{10}{3} - \frac{2}{3}y - \frac{4}{3}z, \quad 3 - \frac{2}{3}z, \quad \frac{5}{2} + \frac{1}{2}y \right\} \right].$$

Note that this method, which is called **Fourier-Motzkin elimination**, is in general very inefficient. If  $m$  is the number of inequalities in the initial system, it may be necessary to state  $\frac{m^2}{4}$  inequalities in the system with one variable less (this is the case if there are  $\frac{m}{2}$  inequalities that gave an upper bound on the variable we got rid of and  $\frac{m}{2}$  inequalities that gave a lower bound).

Nevertheless, the Fourier-Motzkin elimination can be used to get a certificate that a given system of inequalities does *not* have a feasible solution. In the proof of the following theorem we give a general description of one iteration of the method:

**Theorem 4** *Let  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$  (with  $n \geq 1$ ). Then there are  $\tilde{A} \in \mathbb{R}^{\tilde{m} \times (n-1)}$  and  $\tilde{b} \in \mathbb{R}^{\tilde{m}}$  with  $\tilde{m} \leq \max\{m, \frac{m^2}{4}\}$  such that*

- (a) *Each inequality in the system  $\tilde{A}\tilde{x} \leq \tilde{b}$  is a positive linear combination of inequalities from  $Ax \leq b$*
- (b) *The system  $Ax \leq b$  has a solution if and only if  $\tilde{A}\tilde{x} \leq \tilde{b}$  has a solution.*

**Proof:** Denote the entries of  $A$  by  $a_{ij}$ , i.e.  $A = (a_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}}$ . We will show how to get rid of the variable with index 1. To this end, we partition the index set  $\{1, \dots, m\}$  of the rows into three disjoint sets  $U, L$ , and  $N$ :

$$\begin{aligned} U &:= \{i \in \{1, \dots, m\} \mid a_{i1} > 0\} \\ L &:= \{i \in \{1, \dots, m\} \mid a_{i1} < 0\} \\ N &:= \{i \in \{1, \dots, m\} \mid a_{i1} = 0\} \end{aligned}$$

We can assume that  $|a_{i1}| = 1$  for all  $i \in U \cup L$  (otherwise we divide the corresponding inequality by  $|a_{i1}|$ ).

For vectors  $\tilde{a}_i = (a_{i2}, \dots, a_{in})$  and  $\tilde{x} = (x_2, \dots, x_n)$  (that are empty if  $n = 1$ ), we replace the inequalities that correspond to indices in  $U$  and  $L$  by

$$\tilde{a}_i^t \tilde{x} + \tilde{a}_k^t \tilde{x} \leq b_i + b_k \quad i \in U, k \in L. \quad (17)$$

Obviously, each of these  $|U| \cdot |L|$  new inequalities is simply the sum of two of the given inequalities (and hence a positive linear combination of them).

The inequalities with index in  $N$  are rewritten as

$$\tilde{a}_l^t \tilde{x} \leq b_l \quad l \in N. \quad (18)$$

The inequalities in (17) and (18) form a set of inequalities  $\tilde{A}\tilde{x} \leq \tilde{b}$  with  $n - 1$  variables, and each solution of  $Ax \leq b$  gives a solution of  $\tilde{A}\tilde{x} \leq \tilde{b}$  by restricting  $x = (x_1, \dots, x_n)$  to  $(x_2, \dots, x_n)$ .

On the other hand, if  $\tilde{x} = (x_2, \dots, x_n)$  is a solution of  $\tilde{A}\tilde{x} \leq \tilde{b}$ , then we can set  $\tilde{x}_1$  to any value in the (non-empty) interval

$$\left[ \max\{\tilde{a}_k^t \tilde{x} - b_k \mid k \in L\}, \quad \min\{b_i - \tilde{a}_i^t \tilde{x} \mid i \in U\} \right]$$

where we set the minimum of an empty set to  $\infty$  and the maximum of an empty set to  $-\infty$ . Then,  $x = (\tilde{x}_1, x_2, \dots, x_n)$  is a solution of  $Ax \leq b$ .  $\square$

## 2.3 Farkas' Lemma

**Theorem 5** (*Farkas' Lemma for a system of inequalities*) For  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ , the system  $Ax \leq b$  has a solution if and only if there is no vector  $u \in \mathbb{R}^m$  with  $u \geq 0$ ,  $u^t A = 0^t$  and  $u^t b < 0$ .

**Proof:** “ $\Rightarrow$ .” If  $Ax \leq b$  and  $u \in \mathbb{R}^m$  with  $u \geq 0$ ,  $u^t A = 0^t$  and  $u^t b < 0$ , then  $0 = (u^t A)x = u^t(Ax) \leq u^t b < 0$ , which is a contradiction.

“ $\Leftarrow$ .” Assume that  $Ax \leq b$  does not have a solution. Let  $A^{(0)} := A$  and  $b^{(0)} := b$ . We apply Theorem 4 to  $A^{(0)}x^{(0)} \leq b^{(0)}$  and get a system  $A^{(1)}x^{(1)} \leq b^{(1)}$  of inequalities with  $n - 1$  variables such that  $A^{(1)}x^{(1)} \leq b^{(1)}$  does not have a solution either and such that each inequality of  $A^{(1)}x^{(1)} \leq b^{(1)}$  is a positive linear combination of inequalities of  $A^{(0)}x^{(0)} \leq b^{(0)}$ . We iterate this step  $n$  times, and in the end, we get a system of inequalities  $A^{(n)}x^{(n)} \leq b^{(n)}$  without variables (so  $x^{(n)}$  is in fact a vector of length 0) that does not have a solution. Moreover, each inequality in  $A^{(n)}x^{(n)} \leq b^{(n)}$  is a positive linear combination of inequalities in  $Ax \leq b$ . Since  $A^{(n)}x^{(n)} \leq b^{(n)}$  does not have a solution, it must contain an inequality  $0 \leq d$  for a constant  $d < 0$ . This is a positive linear combination of inequalities in  $Ax \leq b$ , so there is a vector  $u \in \mathbb{R}^m$  with  $u \geq 0$ ,  $u^t A = 0^t$  and  $u^t b = d < 0$ .  $\square$

**Theorem 6** (*Farkas' Lemma, most general case*) For  $A \in \mathbb{R}^{m_1 \times n_1}$ ,  $B \in \mathbb{R}^{m_1 \times n_2}$ ,  $C \in \mathbb{R}^{m_2 \times n_1}$ ,  $D \in \mathbb{R}^{m_2 \times n_2}$ ,  $a \in \mathbb{R}^{m_1}$  and  $b \in \mathbb{R}^{m_2}$  exactly one of the two following systems has a feasible solution:

System 1:

$$\begin{aligned} Ax + By &\leq a \\ Cx + Dy &= b \\ x &\geq 0 \end{aligned} \tag{19}$$

System 2:

$$\begin{aligned} u^t A + v^t C &\geq 0^t \\ u^t B + v^t D &= 0^t \\ u &\geq 0 \\ u^t a + v^t b &< 0 \end{aligned} \tag{20}$$

**Proof:** The first system is equivalent to

$$\begin{aligned} Ax + By &\leq a \\ Cx + Dy &\leq b \\ -Cx - Dy &\leq -b \\ -I_{n_1}x &\leq 0 \end{aligned}$$

By Theorem 5, this system has a solution if and only if the following system does not have a solution:

$$\begin{aligned} u_1^t A + u_2^t C - u_3^t C - u_4^t &= 0^t \\ u_1^t B + u_2^t D - u_3^t D &= 0^t \\ u_1^t a + u_2^t b - u_3^t b &< 0^t \\ u_1 &\geq 0 \\ &u_2 \geq 0 \\ &u_3 \geq 0 \\ &u_4 \geq 0 \end{aligned}$$

Obviously, this system has a solution if and only if the second system of the theorem has a solution.  $\square$

**Corollary 7** (*Farkas' Lemma, further variants*) For  $A \in \mathbb{R}^{m \times n}$  and  $b \in \mathbb{R}^m$ , the following statements hold:

- (a) There is a vector  $x \in \mathbb{R}^n$  with  $x \geq 0$  and  $Ax = b$  if and only if there is no vector  $u \in \mathbb{R}^m$  with  $u^t A \geq 0^t$  and  $u^t b < 0$ .
- (b) There is a vector  $x \in \mathbb{R}^n$  with  $Ax \leq b$  if and only if there is no vector  $u \in \mathbb{R}^m$  with  $u^t A = 0^t$  and  $u^t b < 0$ .

**Proof:** Restrict the statement of Theorem 6 to the vector  $b$  and matrix  $C$  (for part (a)) of  $D$  (for part (b)).  $\square$

**Remark:** Statement (a) of Corollary 7 has a nice geometric interpretation. Let  $C$  be the cone generated by the columns of  $A$ . Then, the vector  $b$  is either in  $C$  or there is a hyperplane (given by the normal  $u$ ) that separates  $b$  from  $C$ .

As an example consider  $A = \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix}$  and  $b_1 = \begin{pmatrix} 5 \\ 2 \end{pmatrix}$  and  $b_2 = \begin{pmatrix} 1 \\ 3 \end{pmatrix}$  (see Figure 2). The vector  $b_1$  is in the cone generated by the columns of  $A$  (because  $\begin{pmatrix} 5 \\ 2 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix} + \begin{pmatrix} 3 \\ 1 \end{pmatrix}$ ) while  $b_2$  can be separated from the cone by a hyperplane orthogonal to  $u = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$ .

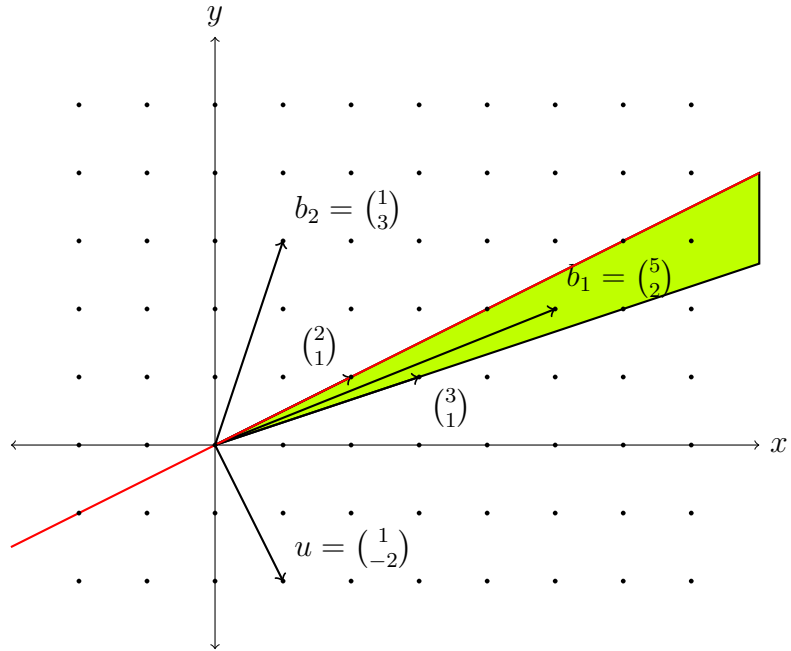


Fig. 2: Example for the statement in Corollary 7(a).

## 2.4 Strong Duality

**Theorem 8** (*Strong duality*) For the two linear programs

$$\begin{aligned} \max c^t x & & (P) \\ \text{s.t. } Ax & \leq b \end{aligned}$$

and

$$\begin{aligned} \min b^t y & & (D) \\ \text{s.t. } A^t y & = c \\ y & \geq 0 \end{aligned}$$

exactly one of the following statements is true:

1. Neither (P) nor (D) have a feasible solution.
2. (P) is unbounded and (D) has no feasible solution.
3. (P) has no feasible solution and (D) is unbounded.
4. Both (P) and (D) have a feasible solution. Then both have an optimal solution, and for an optimal solution  $\tilde{x}$  of (P) and an optimal solution  $\tilde{y}$  of (D), we have

$$c^t \tilde{x} = b^t \tilde{y}.$$

**Proof:** Obviously, at most one of the statements can be true.

If one of the linear programs is unbounded, then the other one must be infeasible because of the weak duality.

Assume that one of the LPs (say (P) without loss of generality) is feasible and bounded.

Hence the system

$$Ax \leq b \tag{21}$$

has a feasible solution while there is a  $B$  such that the system

$$\begin{aligned} Ax & \leq b \\ -c^t x & \leq -B \end{aligned} \tag{22}$$

does not have a feasible solution. By Farkas' Lemma (Theorem 5), this means that there is a vector  $u \in \mathbb{R}^m$  and a number  $z \in \mathbb{R}$  with  $u \geq 0$  and  $z \geq 0$  such that  $u^t A - zc^t = 0^t$  and  $b^t u - zB < 0$ .

Note that  $z > 0$  because if  $z = 0$ , then  $u^t A = 0^t$  and  $b^t u < 0$  which means that  $Ax \leq b$  does not have a feasible solution, which is a contradiction to our assumption. Therefore, we can define

$\tilde{u} := \frac{1}{z}u$ . This implies  $A^t\tilde{u} = c$  and  $\tilde{u} \geq 0$ , so  $\tilde{u}$  is a feasible solution of (D). Therefore (D) is feasible. It is bounded as well because of the weak duality.

It remains to show that there are feasible solutions  $x$  of (P) and  $y$  of (D) such that  $c^t x \geq b^t y$ .

This is the case if (and only if) the following system has a feasible solution:

$$\begin{array}{rcl} Ax & \leq & b \\ & A^t y & = c \\ -c^t x + b^t y & \leq & 0 \\ & y & \geq 0 \end{array}$$

By Theorem 6, this is the case if and only if the following system (with variables  $u \in \mathbb{R}^m$ ,  $v \in \mathbb{R}^n$  and  $w \in \mathbb{R}$ ) does *not* have a feasible solution:

$$\begin{array}{rcl} u^t A & -w c^t & = 0 \\ & v^t A^t + w b^t & \geq 0 \\ u^t b + v^t c & & < 0 \\ u & & \geq 0 \\ & w & \geq 0 \end{array} \tag{23}$$

Hence, assume that system (23) has a feasible solution  $u$ ,  $v$  and  $w$ .

**Case 1:**  $w = 0$ . Then (again by Farkas' Lemma) the system

$$\begin{array}{rcl} Ax & \leq & b \\ & A^t y & = c \\ & y & \geq 0 \end{array}$$

does not have a feasible solution, which is a contradiction because both (P) and (D) have a feasible solution.

**Case 2:**  $w > 0$ . Then

$$0 > w u^t b + w v^t c \geq u^t (-A v) + v^t (A^t u) = 0,$$

which is a contradiction. □

**Remark:** Theorem 8 shows in particular that if a linear program  $\max\{c^t x \mid Ax \leq b\}$  is feasible and unbounded that there is a vector  $\tilde{x}$  with  $A\tilde{x} \leq b$  such that  $c^t \tilde{x} = \sup\{c^t x \mid Ax \leq b\}$ .

The following table gives an overview of the possible combinations of states of the primal and dual LPs (“✓” means that the combination is possible, “x” means that it is not possible):

		(D)		
		Feasible, bounded	Feasible, unbounded	Infeasible
(P)	Feasible, bounded	✓	x	x
	Feasible, unbounded	x	x	✓
	Infeasible	x	✓	✓

**Remark:** The previous theorem can be used to show that computing a feasible solution of a linear program is in general as hard as computing an optimum solution. Assume that we want to compute an optimum solution of the program (P) in the theorem. To this end, we can compute any feasible solution of the following linear program:

$$\begin{aligned}
 & \max c^t x \\
 \text{s.t. } & Ax \leq b \\
 & A^t y = c \\
 & c^t x \geq b^t y \\
 & y \geq 0
 \end{aligned} \tag{24}$$

Here  $x$  and  $y$  are the variables. We can ignore the objective function in the modified LP because we just need any feasible solution. The constraints  $A^t y = c$ ,  $c^t x \geq b^t y$  and  $y \geq 0$  guarantee that any vector  $x$  from a feasible solution of the new LP is an optimum solution of (P).

**Corollary 9** *Let  $A, B, C, D, E, F, G, H, K$  be matrices and  $a, b, c, d, e, f$  be vectors of appropriate dimensions such that:*

$$\begin{pmatrix} A & B & C \\ D & E & F \\ G & H & K \end{pmatrix} \text{ is an } m \times n\text{-matrix,}$$

$$\begin{pmatrix} a \\ b \\ c \end{pmatrix} \text{ is a vector of length } m \text{ and } \begin{pmatrix} d \\ e \\ f \end{pmatrix} \text{ is a vector of length } n.$$

Then

$$\begin{aligned}
 & \max \left\{ \begin{array}{l} Ax + By + Cz \leq a \\ Dx + Ey + Fz = b \\ d^t x + e^t y + f^t z : \begin{array}{l} Gx + Hy + Kz \geq c \\ x \geq 0 \\ z \leq 0 \end{array} \end{array} \right\} \\
 & = \\
 & \min \left\{ \begin{array}{l} A^t u + D^t v + G^t w \geq d \\ B^t u + E^t v + H^t w = e \\ a^t u + b^t v + c^t w : \begin{array}{l} C^t u + F^t v + K^t w \leq f \\ u \geq 0 \\ w \leq 0 \end{array} \end{array} \right\},
 \end{aligned}$$

*provided that both sets are non-empty.*

**Proof:** Transform the first LP into standard inequality form and apply Theorem 8. The details are again left as an exercise.  $\square$

Table 1 gives an overview of how a primal linear program can be converted into a dual linear program.



	Primal LP	Dual LP
Variables	$x_1, \dots, x_n$	$y_1, \dots, y_m$
Matrix	$A$	$A^t$
Right-hand side	$b$	$c$
Objective function	$\max c^t x$	$\min b^t y$
Constraints	$\sum_{j=1}^n a_{ij} x_j \leq b_i$	$y_i \geq 0$
	$\sum_{j=1}^n a_{ij} x_j \geq b_i$	$y_i \leq 0$
	$\sum_{j=1}^n a_{ij} x_j = b_i$	$y_i \in \mathbb{R}$
	$x_j \geq 0$	$\sum_{i=1}^m a_{ij} y_i \geq c_j$
	$x_j \leq 0$	$\sum_{i=1}^m a_{ij} y_i \leq c_j$
	$x_j \in \mathbb{R}$	$\sum_{i=1}^m a_{ij} y_i = c_j$

Tabelle 1: Dualization of linear programs.

Here are some important special cases of primal-dual pairs of LPs:

Primal LP	Dual LP
$\max\{c^t x \mid Ax \leq b\}$	$\min\{b^t y \mid y^t A = c, y \geq 0\}$
$\max\{c^t x \mid Ax \leq b, x \geq 0\}$	$\min\{b^t y \mid y^t A \geq c, y \geq 0\}$
$\max\{c^t x \mid Ax \geq b, x \geq 0\}$	$\min\{b^t y \mid y^t A \geq c, y \leq 0\}$
$\max\{c^t x \mid Ax = b, x \geq 0\}$	$\min\{b^t y \mid y^t A \geq c\}$

## 2.5 Complementary Slackness

**Theorem 10** (*Complementary slackness for inequalities*) Let  $\max\{c^t x \mid Ax \leq b\}$  and  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  be a pair of a primal and a dual linear program. Then, for  $x \in \mathbb{R}^n$  with  $Ax \leq b$  and  $y \in \mathbb{R}^m$  with  $A^t y = c$  and  $y \geq 0$  the following statements are equivalent:

- (a)  $x$  is an optimum solution of  $\max\{c^t x \mid Ax \leq b\}$  and  $y$  an optimum solution of  $\min\{b^t y \mid A^t y = c, y \geq 0\}$ .
- (b)  $c^t x = b^t y$ .
- (c)  $y^t(b - Ax) = 0$ .

**Proof:** The equivalence of the statements (a) and (b) follows from Theorem 8. To see the equivalence of (b) and (c) note that  $y^t(b - Ax) = y^tb - y^tAx = y^tb - c^tx$ , so  $c^tx = b^ty$  is equivalent to  $y^t(b - Ax) = 0$ .  $\square$

With the notation of the theorem, let  $a_1^t, \dots, a_m^t$  be the rows of  $A$  and  $b = (b_1, \dots, b_m)$ . Then, the theorem implies that for an optimum primal solution  $x$  and an optimum dual solution  $y$  and  $i \in \{1, \dots, m\}$  we have  $y_i = 0$  or  $a_i^tx = b_i$  (since  $\sum_{i=1}^m y_i(b_i - a_i^tx)$  must be zero and  $y_i(b_i - a_i^tx)$  cannot be negative for any  $i \in \{1, \dots, m\}$ ).

**Theorem 11** (*Complementary slackness for inequalities with non-negative variables*) Let  $\max\{c^tx \mid Ax \leq b, x \geq 0\}$  and  $\min\{b^ty \mid A^ty \geq c, y \geq 0\}$  be a pair of a primal and a dual linear program. Then, for  $x \in \mathbb{R}^n$  with  $Ax \leq b$  and  $x \geq 0$  and  $y \in \mathbb{R}^m$  with  $A^ty \geq c$  and  $y \geq 0$  the following statements are equivalent:

(a)  $x$  is an optimum solution of  $\max\{c^tx \mid Ax \leq b, x \geq 0\}$  and  $y$  an optimum solution of  $\min\{b^ty \mid A^ty \geq c, y \geq 0\}$ .

(b)  $c^tx = b^ty$ .

(c)  $y^t(b - Ax) = 0$  and  $x^t(A^ty - c) = 0$ .

**Proof:** The equivalence of the statements (a) and (b) follows again from Theorem 8. To see the equivalence of (b) and (c) note that  $0 \leq y^t(b - Ax)$  and  $0 \leq x^t(A^ty - c)$ . Hence  $y^t(b - Ax) + x^t(A^ty - c) = y^tb - y^tAx + x^tA^ty - x^tc = y^tb - x^tc$  is zero if and only if  $0 = y^t(b - Ax)$  and  $0 = x^t(A^ty - c)$ .  $\square$

**Corollary 12** Let  $\max\{c^tx \mid Ax \leq b\}$  be a feasible linear program. Then, the linear program is bounded if and only if  $c$  is in the convex cone generated by the rows of  $A$ .

**Proof:** The linear program is bounded if and only if its dual linear program is feasible. This is the case if and only if there is a vector  $y \geq 0$  with  $y^tA = c$  which is equivalent to the statement that  $c$  is in the cone generated by the rows of  $A$ .  $\square$

Theorem 10 allows us to strengthen the statement of the previous Corollary. Let  $x$  be an optimum solution of the linear program  $\max\{c^tx \mid Ax \leq b\}$  and  $y$  an optimum solution of its dual  $\min\{b^ty \mid A^ty = c, y \geq 0\}$ . Denote the row vectors of  $A$  by  $a_1^t, \dots, a_m^t$ . Then  $y_i = 0$  if  $a_i^tx < b_i$  (for  $i \in \{1, \dots, m\}$ ), so  $c$  is in fact in the cone generated only by those rows of  $A$  where  $a_i^tx = b_i$  (see Figure 3 for an illustration).

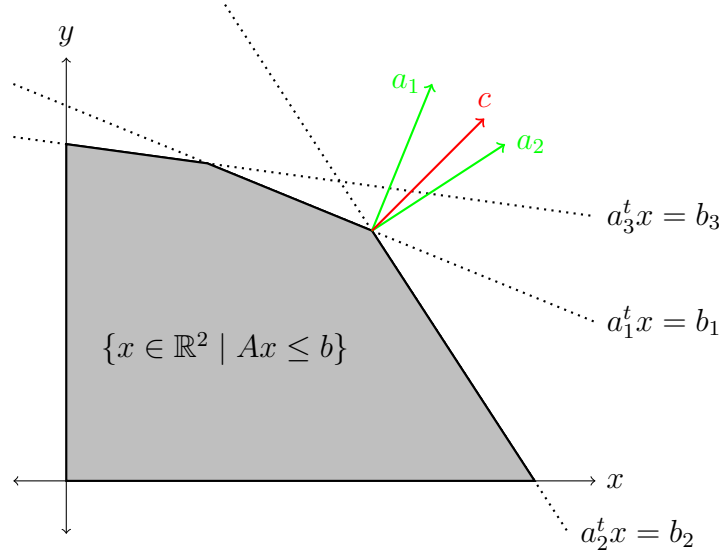


Fig. 3: Cost vector  $c$  as non-negative combination of rows in  $A$ .

**Theorem 13** (*Strict Complementary Slackness*) Let  $\max\{c^t x \mid Ax \leq b\}$  and  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  be a pair of a primal and a dual linear program that are both feasible and bounded. Then, for each inequality  $a_i^t x \leq b_i$  in  $Ax \leq b$  exactly one of the following two statements holds:

- (a) The primal LP  $\max\{c^t x \mid Ax \leq b\}$  has an optimum solution  $x^*$  with  $a_i^t x^* < b_i$ .
- (b) The dual LP  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  has an optimum solution  $y^*$  with  $y_i^* > 0$ .

**Proof:** By complementary slackness, at most one the statements can be true. Let  $\delta = \max\{c^t x \mid Ax \leq b\}$  be the value of an optimum solution. Assume that (a) does not hold. This means that

$$\begin{aligned} \max \quad & -a_i^t x \\ & Ax \leq b \\ & -c^t x \leq -\delta \end{aligned}$$

has an optimum solution with  $-b_i$ . Hence, also its dual LP

$$\begin{aligned} \min \quad & b^t y - \delta u \\ & A^t y - uc = -a_i \\ & y \geq 0 \\ & u \geq 0 \end{aligned}$$

must have an optimum solution of value  $-b_i$ . Therefore, there are  $y \in \mathbb{R}^m$  and  $u \in \mathbb{R}$  with  $y \geq 0$  and  $u \geq 0$  with  $y^t A - uc^t = -a_i^t$  and  $y^t b - u\delta = -b_i$ . Let  $\tilde{y} = y + e_i$  (i.e.  $\tilde{y}$  arises from  $y$  by increasing the  $i$ -th entry by one). If  $u = 0$ , then  $\tilde{y}^t A = y^t A + a_i^t = 0$  and  $\tilde{y}^t b = y^t b + b_i = 0$ , so if  $y^*$

is an optimum solution of  $\min\{b^t y \mid A^t y = c, y \geq 0\}$ , then  $y^* + \tilde{y}$  is also an optimum solution and has a positive  $i$ -th entry. If  $u > 0$ , then  $\frac{1}{u}\tilde{y}$  is an optimum solution of  $\min\{b^t y \mid A^t y = c, y \geq 0\}$ , (because  $\frac{1}{u}\tilde{y}^t A = \frac{1}{u}y^t A + \frac{1}{u}a_i^t = c^t$  and  $\frac{1}{u}\tilde{y}^t b = \frac{1}{u}y^t b + \frac{1}{u}b_i = \delta$ ) and has a positive  $i$ -th entry.  $\square$

**Theorem 14** *Let  $\max\{c^t x \mid Ax \leq b\}$  and  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  be a pair of a primal and a dual linear program that are both feasible and bounded. Then, there are optimum solution  $x^*$  and  $y^*$  of the LPs such that for each inequality  $a_i^t x \leq b_i$  in  $Ax \leq b$  either  $a_i^t x^* < b_i$  or  $y_i^* > 0$  holds.*

**Proof:** By Theorem 13, for any inequality  $a_i^t x \leq b_i$  there is a pair of optimum solutions  $x^{(i)} \in \mathbb{R}^n$ ,  $y^{(i)} \in \mathbb{R}^m$  such that  $a_i^t x^{(i)} < b_i$  or  $y_i^{(i)} > 0$ . Since the convex combination of optimum LP solutions is again an optimum solution, we can set  $x^* := \frac{1}{m} \sum_{i=1}^m x^{(i)}$  and  $y^* := \frac{1}{m} \sum_{i=1}^m y^{(i)}$  and get a pair of optimum solutions fulfilling the conditions of the theorem.  $\square$

As an application of complementary slackness we consider again the MAXIMUM-FLOW PROBLEM. Let  $G$  be a directed graph with  $s, t \in V(G)$ ,  $s \neq t$ , and capacities  $u : E(G) \rightarrow \mathbb{R}_{>0}$ . Here is the LP-formulation of the MAXIMUM-FLOW PROBLEM:

$$\begin{aligned}
\max \quad & \sum_{e \in \delta_G^+(s)} x_e - \sum_{e \in \delta_G^-(s)} x_e \\
\text{s.t.} \quad & x_e \geq 0 \quad \text{for } e \in E(G) \\
& x_e \leq u(e) \quad \text{for } e \in E(G) \\
& \sum_{e \in \delta_G^+(v)} x_e - \sum_{e \in \delta_G^-(v)} x_e = 0 \quad \text{for } v \in V(G) \setminus \{s, t\}
\end{aligned} \tag{25}$$

By dualizing it, we get

$$\begin{aligned}
\min \quad & \sum_{e \in E(G)} u(e)y_e \\
\text{s.t.} \quad & y_e \geq 0 \quad \text{for } e \in E(G) \\
& y_e + z_v - z_w \geq 0 \quad \text{for } e = (v, w) \in E(G), \{s, t\} \cap \{v, w\} = \emptyset \\
& y_e + z_v \geq 0 \quad \text{for } e = (v, t) \in E(G), v \neq s \\
& y_e - z_w \geq 0 \quad \text{for } e = (t, w) \in E(G), w \neq s \\
& y_e - z_w \geq 1 \quad \text{for } e = (s, w) \in E(G), w \neq t \\
& y_e + z_v \geq -1 \quad \text{for } e = (v, s) \in E(G), v \neq t \\
& y_e \geq 1 \quad \text{for } e = (s, t) \in E(G) \\
& y_e \geq -1 \quad \text{for } e = (t, s) \in E(G)
\end{aligned} \tag{26}$$

In a simplified way its dual LP can be written with two dummy variables  $z_s = -1$  and  $z_t = 0$ :

$$\begin{aligned}
\min \quad & \sum_{e \in E(G)} u(e)y_e \\
\text{s.t.} \quad & y_e \geq 0 \quad \text{for } e \in E(G) \\
& y_e + z_v - z_w \geq 0 \quad \text{for } e = (v, w) \in E(G) \\
& z_s = -1 \\
& z_t = 0
\end{aligned} \tag{27}$$

We will use the dual LP to show the Max-Flow-Min-Cut-Theorem. We call a set  $\delta^+(R)$  with  $R \subset V(G)$ ,  $s \in R$  and  $t \notin R$  an  $s$ - $t$ -cut.

**Theorem 15** (*Max-Flow-Min-Cut-Theorem*) *Let  $G$  be a directed graph with edge capacities  $u : E(G) \rightarrow \mathbb{R}_{>0}$ . Let  $s, t \in V(G)$  be two different vertices. Then, the minimum of all capacities of  $s$ - $t$ -cuts equals the maximum value of an  $s$ - $t$ -flow.*

**Proof:** If  $x$  is a feasible solution of the primal problem (25) (i.e.  $x$  encodes an  $s$ - $t$ -flow) and  $\delta^+(R)$  is an  $s$ - $t$ -cut, then

$$\sum_{e \in \delta_G^+(s)} x_e - \sum_{e \in \delta_G^-(s)} x_e = \sum_{v \in R} \left( \sum_{e \in \delta_G^+(v)} x_e - \sum_{e \in \delta_G^-(v)} x_e \right) = \sum_{e \in \delta_G^+(R)} x_e - \sum_{e \in \delta_G^-(R)} x_e \leq \sum_{e \in \delta_G^+(R)} u(e).$$

The first equation follows from the flow conservation rule (i.e.  $\sum_{e \in \delta_G^+(v)} x_e - \sum_{e \in \delta_G^-(v)} x_e = 0$ ) applied to all vertices in  $R \setminus \{s\}$  and the second one from the fact that flow values on edges inside  $R$  cancel out in the sum. The last inequality follows from the fact that flow values are between 0 and  $u$ .

Thus, the capacity of any  $s$ - $t$ -cut is an upper bound for the value of an  $s$ - $t$ -flow. We will show that for any maximum  $s$ - $t$ -flow there is an  $s$ - $t$ -cut whose capacity equals the value of the flow.

Let  $\tilde{x}$  be an optimum solution of the primal problem (25) and  $\tilde{y}, \tilde{z}$  be an optimum solution of the dual problem (27). In particular  $\tilde{x}$  defines a maximum  $s$ - $t$ -flow. Consider the set  $R := \{v \in V(G) \mid \tilde{z}_v \leq -1\}$ . Then  $s \in R$  and  $t \notin R$ .

If  $e = (v, w) \in \delta_G^+(R)$ , then  $\tilde{z}_v < \tilde{z}_w$ , so  $\tilde{y}_e \geq \tilde{z}_w - \tilde{z}_v > 0$ . By complementary slackness this implies  $\tilde{x}_e = u(e)$ . On the other hand, if  $e = (v, w) \in \delta_G^-(R)$ , then  $\tilde{z}_v > \tilde{z}_w$  and hence  $\tilde{y}_e + \tilde{z}_v - \tilde{z}_w \geq \tilde{z}_v - \tilde{z}_w > 0$ , so again by complementary slackness  $\tilde{x}_e = 0$ . This leads to:

$$\sum_{e \in \delta_G^+(s)} \tilde{x}_e - \sum_{e \in \delta_G^-(s)} \tilde{x}_e = \sum_{v \in R} \left( \sum_{e \in \delta_G^+(v)} \tilde{x}_e - \sum_{e \in \delta_G^-(v)} \tilde{x}_e \right) = \sum_{e \in \delta_G^+(R)} \tilde{x}_e - \sum_{e \in \delta_G^-(R)} \tilde{x}_e = \sum_{e \in \delta_G^+(R)} u(e).$$

□



### 3 The Structure of Polyhedra

#### 3.1 Mappings of Polyhedra

**Proposition 16** Let  $A \in \mathbb{R}^{m \times (n+k)}$  and  $b \in \mathbb{R}^m$ . Then the set

$$P = \{x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^k : A \begin{pmatrix} x \\ y \end{pmatrix} \leq b\}$$

is a polyhedron.

**Proof:** Exercise. □

**Remark:** The set  $P = \{x \in \mathbb{R}^n \mid \exists y \in \mathbb{R}^k : A \begin{pmatrix} x \\ y \end{pmatrix} \leq b\}$  is called a **projection** of  $\{z \in \mathbb{R}^{n+k} \mid Az \leq b\}$  to  $\mathbb{R}^n$ .

More generally, the image of a polyhedron  $\{x \in \mathbb{R}^n \mid Ax \leq b\}$  under an **affine linear mapping**  $f : \mathbb{R}^n \rightarrow \mathbb{R}^k$ , which is given by  $D \in \mathbb{R}^{k \times n}$ ,  $d \in \mathbb{R}^k$  and  $x \mapsto Dx + d$  is also a polyhedron:

**Corollary 17** Let  $A \in \mathbb{R}^{m \times n}$ ,  $b \in \mathbb{R}^m$ ,  $D \in \mathbb{R}^{k \times n}$  and  $d \in \mathbb{R}^k$ . Then

$$\{y \in \mathbb{R}^k \mid \exists x \in \mathbb{R}^n : Ax \leq b \text{ and } y = Dx + d\}$$

is a polyhedron.

**Proof:** Note that

$$\begin{aligned} & \left\{ y \in \mathbb{R}^k \mid \exists x \in \mathbb{R}^n : Ax \leq b \text{ and } y = Dx + d \right\} \\ &= \left\{ y \in \mathbb{R}^k \mid \exists x \in \mathbb{R}^n \left( \begin{pmatrix} A & 0 \\ D & -I_k \\ -D & I_k \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \leq \begin{pmatrix} b \\ -d \\ d \end{pmatrix} \right) \right\} \end{aligned}$$

and apply the previous proposition. □

## 3.2 Faces

**Definition 9** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a non-empty polyhedron and  $c \in \mathbb{R}^n \setminus \{0\}$ .

- (a) For  $\delta := \max\{c^t x \mid x \in P\} < \infty$ , the set  $\{x \in \mathbb{R}^n \mid c^t x = \delta\}$  is called **supporting hyperplane** of  $P$ .
- (b) A set  $X \subseteq \mathbb{R}^n$  is called **face** of  $P$  if  $X = P$  or if there is a supporting hyperplane  $H$  of  $P$  such that  $X = P \cap H$ .
- (c) If  $\{x'\}$  is a face of  $P$ , we call  $x'$  **vertex** of  $P$  or **basic solution** of the system  $Ax \leq b$ .

**Proposition 18** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a non-empty polyhedron and  $F \subseteq P$ . Then, the following statements are equivalent:

- (a)  $F$  is a face of  $P$ .
- (b) There is a vector  $c \in \mathbb{R}^n$  such that  $\delta := \max\{c^t x \mid x \in P\} < \infty$  and  $F = \{x \in P \mid c^t x = \delta\}$ .
- (c) There is a subsystem  $A'x \leq b'$  of  $Ax \leq b$  such that  $F = \{x \in P \mid A'x = b'\} \neq \emptyset$ .

**Proof:**

“(a)  $\Rightarrow$  (b)” : Let  $F$  be face of  $P$ . If  $F = P$ , then  $c = 0$  yields  $F = \{x \in P \mid c^t x = 0\}$ . If  $F \neq P$ , then there must be a  $c \in \mathbb{R}^n$  such that for  $\delta := \max\{c^t x \mid x \in P\} (< \infty)$  we have  $F = \{x \in \mathbb{R}^n \mid c^t x = \delta\} \cap P = \{x \in P \mid c^t x = \delta\}$ .

“(b)  $\Rightarrow$  (c)” : Let  $c, \delta$  and  $F$  be as described in (b). Let  $A'x \leq b'$  be a maximal subsystem of  $Ax \leq b$  such that  $A'x = b'$  for all  $x \in F$ . Hence  $F \subseteq \{x \in P \mid A'x = b'\}$  and it remains to show that  $F \supseteq \{x \in P \mid A'x = b'\}$ . Let  $\tilde{A}x \leq \tilde{b}$  be the inequalities in  $Ax \leq b$  that are not contained in  $A'x \leq b'$ . Denote the inequalities of  $\tilde{A}x \leq \tilde{b}$  by  $\tilde{a}_j^t x \leq \tilde{b}_j$  ( $j = 1, \dots, k$ ). Hence, for each  $j = 1, \dots, k$  we have an  $x_j \in F$  with  $\tilde{a}_j^t x_j < \tilde{b}_j$ .

If  $k > 0$ , we set  $x^* := \frac{1}{k} \sum_{j=1}^k x_j$ . Otherwise let  $x^*$  be an arbitrary element of  $F$ . In any case, we have  $\tilde{a}_j^t x^* < \tilde{b}_j$  for all  $j \in \{1, \dots, k\}$ .

Consider an arbitrary  $y \in P \setminus F$ . We have to show that  $A'y \neq b'$ .

Because of  $y \in P \setminus F$  we know that  $c^t y < \delta$ .

Choose  $\epsilon > 0$  with  $\epsilon < \frac{\tilde{b}_j - \tilde{a}_j^t x^*}{\tilde{a}_j^t (x^* - y)}$  for all  $j \in \{1, \dots, k\}$  with  $\tilde{a}_j^t x^* > \tilde{a}_j^t y$  (note that all these upper bounds on  $\epsilon$  are positive).



Set  $z := x^* + \epsilon(x^* - y)$  (see Figure 4). Then  $c^t z > \delta$ , so  $z \notin P$ . Therefore, there must be an inequality  $a^t x \leq \beta$  of the system  $Ax \leq b$  such that  $a^t z > \beta$ . We claim that this inequality cannot belong to  $\tilde{A}x \leq \tilde{b}$ . To see this assume that  $a^t x \leq \beta$  belongs to  $\tilde{A}x \leq \tilde{b}$ . If  $a^t x^* \leq a^t y$  then

$$a^t z = a^t x^* + \epsilon a^t (x^* - y) \leq a^t x^* < \beta.$$

But if  $a^t x^* > a^t y$  then

$$a^t z = a^t x^* + \epsilon a^t (x^* - y) < a^t x^* + \frac{\beta - a^t x^*}{a^t (x^* - y)} a^t (x^* - y) = \beta.$$

In both cases, we get a contradiction, so the inequality  $a^t x \leq \beta$  belongs to  $A'x \leq b'$ . Therefore,  $a^t y = a^t (x^* + \frac{1}{\epsilon}(x^* - z)) = (1 + \frac{1}{\epsilon})\beta - \frac{1}{\epsilon}a^t z < \beta$ , which means that  $A'y \neq b'$ .

“(c)  $\Rightarrow$  (a)” : Let  $A'x \leq b'$  be a subsystem of  $Ax \leq b$  such that  $F = \{x \in P \mid A'x = b'\}$ . Let  $c^t$  be the sum of all row vectors of  $A'$ , and let  $\delta$  be the sum of the entries of  $b'$ . Then,  $c^t x \leq \delta$  for all  $x \in P$  and  $F = P \cap H$  with  $H = \{x \in \mathbb{R}^n \mid c^t x = \delta\}$ .  $\square$

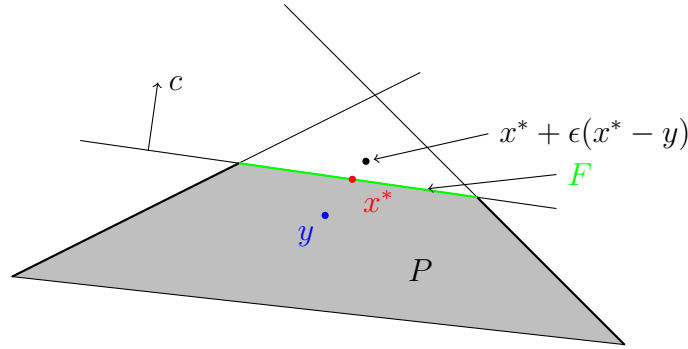


Fig. 4: Illustration of part “(b)  $\Rightarrow$  (c)” of the proof of Proposition 18

The following corollary summarizes direct consequences of the previous proposition:

**Corollary 19** *Let  $P \neq \emptyset$  be a polyhedron and  $F$  a face of  $P$ .*

- (a) *Let  $c \in \mathbb{R}^n$  be a vector such that  $\max\{c^t x \mid x \in P\} < \infty$ . Then the set of all vectors  $x$  where the maximum of  $c^t x$  over  $P$  is attained is a face of  $P$ .*
- (b)  *$F$  is a polyhedron.*
- (c) *A subset  $F' \subseteq F$  is a face of  $F$  if and only if  $F'$  is a face of  $P$ .*
- (d) *If  $P$  is of the form  $P = \{x \in \mathbb{R}^n \mid Ax = b\}$ , so it is in fact a linear subspace, then  $P$  has only one face, namely  $P$  itself.  $\square$*

We are in particular interested in the largest and the smallest faces of a polyhedron.

### 3.3 Facets

**Definition 10** Let  $P$  be a polyhedron. A **facet** of  $P$  is an (inclusion-wise) maximal face  $F$  of  $P$  with  $F \neq P$ . An inequality  $c^t x \leq \delta$  is **facet-defining** for  $P$  if  $c^t x \leq \delta$  for all  $x \in P$  and  $\{x \in P \mid c^t x = \delta\}$  is a facet.

**Theorem 20** Let  $P \subseteq \{x \in \mathbb{R}^n \mid Ax = b\}$  be a non-empty polyhedron of dimension  $n - \text{rank}(A)$ . Let  $A'x \leq b'$  be a minimal system of inequalities such that  $P = \{x \in \mathbb{R}^n \mid Ax = b, A'x \leq b'\}$ . Then, every inequality in  $A'x \leq b'$  is facet-defining for  $P$  and every facet of  $P$  is given by an inequality of  $A'x \leq b'$ .

**Proof:** If  $P = \{x \in \mathbb{R}^n \mid Ax = b\}$ , then  $P$  does not have a facet (the only face of  $P$  is  $P$  itself, see Corollary 19 (d)), so both statements are trivial.

Hence assume that  $P \neq \{x \in \mathbb{R}^n \mid Ax = b\}$ .

Let  $A'x \leq b'$  be a minimal system of inequalities such that  $P = \{x \in \mathbb{R}^n \mid Ax = b, A'x \leq b'\}$ . Let  $a^t x \leq \beta$  be an inequality in  $A'x \leq b'$ , and let  $A''x \leq b''$  be the rest of the system  $A'x \leq b'$  without  $a^t x \leq \beta$ .

We will show that  $a^t x \leq \beta$  is facet-defining.

Let  $y \in \mathbb{R}^n$  be a vector with  $Ay = b$ ,  $A''y \leq b''$  and  $a^t y > \beta$ . Such a vector exists because otherwise  $A''y \leq b''$  would be a smaller system of inequalities than  $A' \leq b'$  with  $P = \{x \in \mathbb{R}^n \mid Ax = b, A'' \leq b''\}$ , which is a contradiction to the definition of  $A'x \leq b'$ .

Moreover, let  $\tilde{y} \in P$  be a vector with  $A'\tilde{y} < b'$  (such a vector  $\tilde{y}$  exists because  $P$  is full-dimensional in the linear subspace  $\{x \in \mathbb{R}^n \mid Ax = b\}$ ). Consider the vector

$$z = \tilde{y} + \frac{\beta - a^t \tilde{y}}{a^t y - a^t \tilde{y}} (y - \tilde{y}).$$

Then,  $a^t z = a^t \tilde{y} + \frac{\beta - a^t \tilde{y}}{a^t y - a^t \tilde{y}} (a^t y - a^t \tilde{y}) = \beta$ . Furthermore,  $0 < \frac{\beta - a^t \tilde{y}}{a^t y - a^t \tilde{y}} < 1$ . Thus,  $z$  is the convex combination of  $\tilde{y}$  and  $y$ , so  $Az = b$  and  $A''z \leq b''$ . Therefore, we have  $z \in P$ .

Set  $F := \{x \in P \mid a^t x = \beta\}$ . Then,  $F \neq \emptyset$  (because  $z \in F$ ), and  $F \neq P$  because  $\tilde{y} \in P \setminus F$ . Hence,  $F$  is a face of  $P$ . It is also a facet because  $a^t x \leq \beta$  is the only inequality of  $A'x \leq b'$  that is met by all elements of  $F$  with equality (e.g. the vector  $z \in F$  fulfills all inequalities in  $A''x \leq b''$  with strict inequality).

On the other hand, by Proposition 18 any facet is defined by an inequality of  $A'x \leq b'$ .  $\square$

**Corollary 21** *Let  $P \subseteq \mathbb{R}^n$  be a polyhedron.*

(a) *Every face  $F$  of  $P$  with  $F \neq P$  is the intersection of facets of  $P$ .*

(b) *The dimension of every facet of  $P$  is  $\dim(P) - 1$ .* □

In particular, this means that the smallest possible representation of a full-dimensional polyhedron  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  is unique (up to swapping inequalities and multiplying inequalities with positive constants). If possible, we want to describe any polyhedron by facet-defining inequalities because according to the Theorem 20, this gives such a smallest possible description of the polyhedron (with respect to the number of inequalities).

### 3.4 Minimal Faces

**Definition 11** *A face  $F$  of a polyhedron  $P$  is called a **minimal face** if there is no face  $F'$  of  $P$  with  $F' \subsetneq F$ .*

**Proposition 22** *Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a polyhedron. A non-empty set  $F \subseteq P$  is a minimal face of  $P$  if and only if there is a subsystem  $A'x \leq b'$  of  $Ax \leq b$  with  $F = \{x \in \mathbb{R}^n \mid A'x = b'\}$ .*

**Proof:** “ $\Rightarrow$ ” Let  $F$  be a minimal face of  $P$ . By Proposition 18, we know that there is a subsystem  $A'x \leq b'$  of  $Ax \leq b$  with  $F = \{x \in P \mid A'x = b'\}$ . Choose  $A'x \leq b'$  maximal with this property. Let  $\tilde{A}x \leq \tilde{b}$  be a minimal subsystem of  $Ax \leq b$  such that  $F = \{x \in \mathbb{R}^n \mid A'x = b', \tilde{A}x \leq \tilde{b}\}$ .

We have to show the following claim:

**Claim:**  $\tilde{A}x \leq \tilde{b}$  is an empty system of inequalities.

**Proof of the Claim:** Assume that  $a^t x \leq \beta$  is an inequality in  $\tilde{A}x \leq \tilde{b}$ . The inequality  $a^t x \leq \beta$  is not redundant, so by Theorem 20,  $F' = \{x \in \mathbb{R}^n \mid A'a = b', \tilde{A}x \leq \tilde{b}, a^t x = \beta\}$  is a facet of  $F$ , and hence, by Corollary 19,  $F'$  is a face of  $P$ . On the other hand, we have  $F' \neq F$ , because  $a^t x = \beta$  is not valid for all elements of  $F$  (otherwise we could have added  $a^t x \leq \beta$  to the set of inequalities  $A'x \leq b'$ ). This is a contradiction to the minimality of  $F$ . This proves the claim.

“ $\Leftarrow$ ” Assume that  $F = \{x \in \mathbb{R}^n \mid A'x = b'\} \subseteq P$  (for a subsystem  $A'x \leq b'$  of  $Ax \leq b$ ) is non-empty.

Then,  $F$  cannot contain a proper subset as a face (see Corollary 19 (d)).

Moreover,  $F = \{x \in \mathbb{R}^n \mid A'x = b'\} = \{x \in P \mid A'x = b'\}$ , so by Proposition 18 the set  $F$  is a face of  $P$ . Since any proper subset of  $F$  that is a face of  $P$  would also be a face of  $F$  and we know that  $F$  does not contain proper subsets as faces,  $F$  is a minimal face of  $P$ .  $\square$

**Corollary 23** *Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a polyhedron. Then the minimal faces of  $P$  have dimension  $n - \text{rank}(A)$ .*

**Proof:** Let  $F$  be a minimal face of  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . By Proposition 22, it can be written as  $F = \{x \in \mathbb{R}^n \mid A'x = b'\}$  for a subsystem  $A'x \leq b'$  of  $Ax \leq b$ . If  $A'$  had smaller rank than  $A$ , we could add a new constraint  $a^t x \leq \beta$  of  $Ax \leq b$  to  $A'x \leq b'$  such that  $a^t$  is linearly independent to all rows of  $A'$ . Then,  $\{x \in \mathbb{R}^n \mid A'x = b', a^t x = \beta\} \subsetneq F$  would be a face of  $F$  and thus a face of  $P$ . This is a contradiction to the minimality of  $F$ . Hence, we can assume that  $\text{rank}(A') = \text{rank}(A)$ .

Therefore,  $\dim(F) = n - \text{rank}(A') = n - \text{rank}(A)$ .  $\square$

**Proposition 24** *Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a polyhedron and  $x' \in P$ . Then, the following statements are equivalent:*

- (a)  $x'$  is a vertex of  $P$ .
- (b) There is a subsystem  $A'x \leq b'$  of  $Ax \leq b$  of  $n$  inequalities such that the rows of  $A'$  are linearly independent and  $\{x'\} = \{x \in P \mid A'x = b'\}$ .
- (c)  $x'$  cannot be written as a convex combination of vectors in  $P \setminus \{x'\}$ .
- (d) There is no non-zero vector  $d \in \mathbb{R}^n$  such that  $\{x' + d, x' - d\} \subseteq P$ .

**Proof:**

“(a)  $\Leftrightarrow$  (b)”: By Proposition 22,  $x'$  is a vertex if and only if there is a subsystem  $A'x \leq b'$  of  $Ax \leq b$  with  $\{x'\} = \{x \in \mathbb{R}^n \mid A'x = b'\}$ . Since  $\{x'\}$  is of dimension 0, this is the case if and only if the statement in (b) holds.

“(b)  $\Rightarrow$  (c)”: Let  $A'x \leq b'$  be a subsystem of  $n$  inequalities of  $Ax \leq b$  such that the rows of  $A'$  are linearly independent and  $\{x'\} = \{x \in P \mid A'x = b'\}$ . Assume that  $x'$  can be written as a convex combination  $\sum_{i=1}^k \lambda_i x^{(i)}$  of vectors  $x^{(i)} \in P \setminus \{x'\}$  (so  $\lambda_i \geq 0$  for  $i \in \{1, \dots, k\}$  and  $\sum_{i=1}^k \lambda_i = 1$ ). If we had  $a^t x^{(i)} < \beta$  for any inequality  $a^t x \leq \beta$  in  $A'x \leq b'$  and  $i \in \{1, \dots, k\}$ , then  $a^t x' = \sum_{i=1}^k \lambda_i a^t x^{(i)} < \beta$ , which is a contradiction. But then, we have  $x^{(i)} \in \{x \in P \mid A'x = b'\} = \{x'\}$  for all  $i \in \{1, \dots, k\}$ , which is a contradiction, too.

“(c)  $\Rightarrow$  (d)”: If  $\{x' + d, x' - d\} \subseteq P$ , then  $x' = \frac{1}{2}((x' + d) + (x' - d))$ , so  $x'$  can be written as a convex combination of vectors in  $P \setminus \{x'\}$ .

“(d)  $\Rightarrow$  (b)”: Let  $A'x \leq b'$  be a maximal subsystem of  $Ax \leq b$  such that  $A'x' = b'$ . Assume that  $A'$  does not contain  $n$  linearly independent rows. Then, there is a vector  $d$  that is orthogonal to all rows in  $A'$ . Hence, for any  $\epsilon > 0$ , we have  $A'(x' + \epsilon d) = A'(x' - \epsilon d) = b'$ . For any inequality  $a^t x \leq \beta$  that is in  $Ax \leq b$  but not in  $A'x \leq b'$ , we have  $a^t x' < \beta$ . Therefore, if  $\epsilon > 0$  is sufficiently small,  $a^t(x' + \epsilon d) \leq \beta$  and  $a^t(x' - \epsilon d) \leq \beta$  are valid for inequalities  $a^t x \leq \beta$  in  $Ax \leq b$  but not in  $A'x \leq b'$ . In other words, we have  $(x' + \epsilon d) \in P$  and  $(x' - \epsilon d) \in P$ .  $\square$

**Definition 12** A polyhedron is called **pointed** if it is empty or all minimal faces of it are of dimension 0.

**Examples:**

- Polytopes are pointed.

To see this, consider a non-empty polytope  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . If  $\text{rank}(A) < n$ , then there is a vector  $\tilde{x} \in \mathbb{R}^n$  such that  $A\tilde{x} = 0$ . But then for any  $x \in P$  and  $K \in \mathbb{R}$ , we have  $x + K\tilde{x} \in P$ , which is a contradiction to the assumption that  $P$  fits into a ball of finite radius. Hence we have  $\text{rank}(A) = n$ , so  $P$  is pointed.

- Polyhedra  $P$  that can be written as  $P = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$  are pointed.

This can be seen by writing  $P$  as  $P = \{x \in \mathbb{R}^n \mid \begin{pmatrix} A \\ -A \\ -I_n \end{pmatrix} x \leq \begin{pmatrix} b \\ -b \\ 0 \end{pmatrix}\}$ . Obviously, the

matrix  $\begin{pmatrix} A \\ -A \\ -I_n \end{pmatrix}$  has rank  $n$ , hence  $P$  is pointed.

**Corollary 25** If the linear program  $\max\{c^t x \mid Ax \leq b\}$  is feasible and bounded and the polyhedron  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  is pointed, then there is a vertex  $x'$  of  $P$  such that  $c^t x' = \max\{c^t x \mid Ax \leq b\}$ .  $\square$

**Theorem 26** (Carathéodory's Theorem) If  $X \subseteq \mathbb{R}^n$  is a finite set of vectors and  $c \in \text{cone}(X)$  then there are linearly independent vectors  $a_1, \dots, a_k \in X$  such that  $c \in \text{cone}(\{a_1, \dots, a_k\})$ .

**Proof:** Let  $\{a_1, \dots, a_k\}$  be an inclusion-wise minimal set of vectors in  $X$  such that  $c \in \text{cone}(\{a_1, \dots, a_k\})$ . This means that there are positive numbers  $\lambda_1, \dots, \lambda_k$  such that  $c = \sum_{i=1}^k \lambda_i a_i$ .

We show that the vectors  $a_1, \dots, a_k$  are linearly independent. If this is not the case, there are numbers  $\gamma_1, \dots, \gamma_k$  such that  $\sum_{i=1}^k \gamma_i a_i = 0$ . We can assume that at least one  $\gamma_i$  is positive.

Choose  $\sigma$  maximal such that  $\lambda_i - \sigma\gamma_i \geq 0$  for all  $i \in \{1, \dots, k\}$ . Then, in particular, for at least one  $i \in \{1, \dots, k\}$ , we have  $\lambda_i - \sigma\gamma_i = 0$ . Therefore,  $\sum_{i=1}^k (\lambda_i - \sigma\gamma_i)a_i$  is a representation of  $c$  with less vectors, which is a contradiction to the minimality of the set  $\{a_1, \dots, a_k\}$ .  $\square$

**Theorem 27** (*Fundamental Theorem of Linear Inequalities*) Let  $a_1, \dots, a_m, c \in \mathbb{R}^n$  be vectors and let  $t$  be the dimension of the subspace of  $\mathbb{R}^n$  spanned by  $a_1, \dots, a_m, c$  (so  $t$  is the rank of the matrix whose rows are  $a_1^t, \dots, a_m^t, c^t$ ). Then, exactly one of the following statements is true:

- (a)  $c$  can be written as a non-negative combination of linearly independent vectors from  $a_1^t, \dots, a_m^t$ .
- (b) There is a hyperplane  $\{x \in \mathbb{R}^n \mid u^t x = 0\}$  (for a non-zero vector  $u \in \mathbb{R}^n$ ) containing  $t-1$  linearly independent vectors from  $a_1, \dots, a_m$  such that  $a_i^t u \geq 0$  for  $i \in \{1, \dots, m\}$  and  $c^t u < 0$ .

**Proof:** Obviously, at most one of the statements can be valid. Let  $A$  be the matrix with rows  $a_1^t, \dots, a_m^t$ .

If  $c \in \text{cone}(\{a_1, \dots, a_m\})$  then by the previous theorem,  $c$  can be written as a non-negative combination of linearly independent vectors from  $a_1^t, \dots, a_m^t$ .

Hence, assume that  $c \notin \text{cone}(\{a_1, \dots, a_m\})$ , so there is no vector  $v \in \mathbb{R}^m$ ,  $v \geq 0$  such that  $c^t = v^t A$ . By Farkas' Lemma (Theorem 6), this implies that there is a vector  $\tilde{u} \in \mathbb{R}^n$  such that  $A\tilde{u} \geq 0$  and  $c^t \tilde{u} < 0$ . This implies that the following LP (with  $u \in \mathbb{R}^n$  as variable vector) has a feasible solution:

$$\begin{aligned} \max \quad & c^t u \\ \text{s.t.} \quad & c^t u \leq -1 \\ & -c^t u \leq 1 \\ & -Au \leq 0 \end{aligned}$$

Moreover, the LP is bounded (-1 is the value of an optimum solution). Hence, the optimum is attained on a face of the solution polyhedron. By Theorem 22, we can write a minimal face where the optimum solution value is attained as a set  $F = \{u \in \mathbb{R}^n \mid A'u = b'\}$  where  $A'u \leq b'$  is a subsystem of  $c^t u \leq -1$ ,  $-c^t u \leq 1$ ,  $-Au \leq 0$  consisting of  $t$  linearly independent vectors. Hence, any vector  $u \in F$  fulfills the condition of (b).  $\square$

### 3.5 Cones

**Theorem 28** (*Farkas-Minkowski-Weyl Theorem*) A cone is polyhedral if and only if it is finitely generated.

**Proof:** “ $\Leftarrow$ ” Let  $a_1, \dots, a_m \in \mathbb{R}^n$  be vectors. We have to show that  $\text{cone}(\{a_1, \dots, a_m\})$  is polyhedral. W.l.o.g. we can assume that the vectors  $a_1, \dots, a_m$  span the vector space  $\mathbb{R}^n$ . Consider the set  $\mathcal{H}$  of half-spaces  $H_u = \{x \in \mathbb{R}^n \mid u^t x \leq 0\}$  such that for each  $H_u \in \mathcal{H}$  the following conditions hold:

- $\{a_1, \dots, a_m\} \subseteq H_u$ , and
- There are  $n - 1$  linearly independent vectors  $a_{i_1}, \dots, a_{i_{n-1}}$  in  $\{a_1, \dots, a_m\}$  such that  $u^t a_{i_j} = 0$  for  $j \in \{1, \dots, n - 1\}$

The set  $\mathcal{H}$  is finite because there are at most  $\binom{m}{n-1}$  such half-spaces, and by Theorem 27 the set  $\text{cone}(\{a_1, \dots, a_m\})$  is the intersection of these half-spaces. Hence,  $\text{cone}(\{a_1, \dots, a_m\})$  is a polyhedron.

“ $\Rightarrow$ ” Let  $C = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$  be a polyhedral cone. We have to show that  $C$  is finitely generated. Let  $C_A$  be the cone generated by the rows of  $A$ . By the first part of the proof, we know that  $C_A$  (as any other finitely generated cone) is polyhedral. Hence, there are vectors  $d_1, \dots, d_k \in \mathbb{R}^n$  such that  $C_A = \{x \in \mathbb{R}^n \mid d_1^t x \leq 0, \dots, d_k^t x \leq 0\}$ . Let  $C_B = \text{cone}(\{d_1, \dots, d_k\})$  be the cone generated by  $d_1, \dots, d_k$ .

Claim:  $C = C_B$ .

Proof of the claim: “ $C_B \subseteq C$ ”: Every row vector of  $A$  is contained in  $C_A$ . Hence  $Ad_i \leq 0$  for all  $i \in \{1, \dots, k\}$ . Therefore,  $d_i \in C$  (for  $i \in \{1, \dots, k\}$ ) and thus (as  $C$  is a cone)  $C_B \subseteq C$ .

“ $C \subseteq C_B$ ”: Assume that there is a  $y \in C \setminus C_B$ . Again by the first part,  $C_B$  is polyhedral. Thus, there must be a vector  $w \in \mathbb{R}^n$  with  $w^t d_i \leq 0$  (for  $i = 1, \dots, k$ ) and  $w^t y > 0$ . This implies  $w \in C_A$ , and therefore  $w^t x \leq 0$  for all  $x \in C$ . Obviously, together with  $w^t y > 0$  this is a contradiction to the assumption  $y \in C$ .  $\square$

**Remark:** For a set  $S \subseteq \mathbb{R}^n$  we call the set  $S^o = \{x \in \mathbb{R}^n \mid x^t y \leq 0 \text{ for all } y \in S\}$ , the **polar cone** of  $S$  (in particular it obviously is a convex cone). For a polyhedral cone  $C = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$  its polar cone  $C^o$  is the cone generated by the rows of  $A$  (see exercises). We have just seen in the proof that  $C^{oo} = C$  for a polyhedral cone  $C$ .

### 3.6 Polytopes

**Theorem 29** *A set  $X \subseteq \mathbb{R}^n$  is a polytope if and only if it is the convex hull of a finite set of vectors in  $\mathbb{R}^n$ .*

**Proof:** “ $\Rightarrow$ ” Let  $X = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a non-empty polytope. We can write  $X$  as follows:

$$X = \left\{ x \in \mathbb{R}^n \mid \begin{pmatrix} x \\ 1 \end{pmatrix} \in C \right\}$$

where

$$C = \left\{ \begin{pmatrix} x \\ \lambda \end{pmatrix} \in \mathbb{R}^{n+1} \mid \lambda \geq 0, Ax - \lambda b \leq 0 \right\}.$$

The set  $C$  is a polyhedral cone, so by Theorem 28 it is finitely generated by a set  $\begin{pmatrix} x_1 \\ \lambda_1 \end{pmatrix}, \dots, \begin{pmatrix} x_k \\ \lambda_k \end{pmatrix}$  of vectors. Since  $X$  is bounded,  $C$  cannot contain a vector  $\begin{pmatrix} x \\ \lambda \end{pmatrix}$  with non-zero  $x$  but  $\lambda = 0$ . Hence, all  $\lambda_i$  are positive (for  $i \in \{1, \dots, k\}$ ). We can even assume that we have  $\lambda_i = 1$  for all  $i \in \{1, \dots, k\}$  because otherwise we could scale all vectors by the factor  $\lambda_i$ . Thus, we have

$$x \in X \Leftrightarrow \exists \mu_1, \dots, \mu_k \geq 0 : \begin{pmatrix} x \\ 1 \end{pmatrix} = \mu_1 \begin{pmatrix} x_1 \\ 1 \end{pmatrix} + \dots + \mu_k \begin{pmatrix} x_k \\ 1 \end{pmatrix}.$$

This implies that  $X$  is the convex hull of  $x_1, \dots, x_k$ .

“ $\Leftarrow$ ” Let  $X = \text{conv}(\{x_1, \dots, x_k\})$  be the convex hull of  $x_1, \dots, x_k$ . We have to show that  $X$  is a polytope. Let  $C = \text{cone}(\{\begin{pmatrix} x_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} x_k \\ 1 \end{pmatrix}\})$  be the cone generated by  $\begin{pmatrix} x_1 \\ 1 \end{pmatrix}, \dots, \begin{pmatrix} x_k \\ 1 \end{pmatrix}$ .

Then, we have

$$x \in X \Leftrightarrow \begin{pmatrix} x \\ 1 \end{pmatrix} \in C.$$

By Theorem 28,  $C$  is polyhedral, so we can write  $C$  as  $C = \{\begin{pmatrix} x \\ \lambda \end{pmatrix} \mid Ax + b\lambda \leq 0\}$ . This shows  $X = \{x \in \mathbb{R}^n \mid Ax + b \leq 0\}$ , so  $X$  is a polyhedron.

It is even a polytope, because for  $M = \max\{\|x_i\| \mid i \in \{1, \dots, k\}\}$  and  $x \in X$ , we can write  $x$  as  $x = \sum_{i=1}^k \lambda_i x_i$  with  $\lambda_1, \dots, \lambda_k \geq 0$  and  $\sum_{i=1}^k \lambda_i = 1$ , so  $\|x\| \leq \sum_{i=1}^k \lambda_i \|x_i\| \leq M \sum_{i=1}^k \lambda_i = M$ .  $\square$

**Corollary 30** *A polytope is the convex hull of its vertices.*

**Proof:** Let  $P$  be a polytope with vertex set  $X$ . Since  $P$  is convex and  $X \subseteq P$ , we have  $\text{conv}(X) \subseteq P$ . It remains to show that  $P \subseteq \text{conv}(X)$ . Theorem 29 implies that  $\text{conv}(X)$  is a polytope, so in particular a polyhedron. Assume that there is a vector  $y \in P \setminus \text{conv}(X)$ . Then, there is a half-space  $H_y = \{x \in \mathbb{R}^n \mid c^t x \leq \delta\}$  such that  $\text{conv}(X) \subseteq H_y$  and  $y \notin H_y$ . This means that  $c^t y > c^t x$  for all  $x \in X$ , so the maximum of the function  $c^t x$  over  $P$  will not be attained at a vertex. This is a contradiction to Corollary 25.  $\square$

### 3.7 Decomposition of Polyhedra

**Notation:** For two non-empty vector sets  $X, Y \subseteq \mathbb{R}^n$ , we define their **Minkowski sum** as:

$$X + Y := \{z \in \mathbb{R}^n \mid \exists x \in X \exists y \in Y : z = x + y\}.$$

If  $X = \emptyset$ , we define  $X + Y := Y$ , and if  $Y = \emptyset$ , we define  $X + Y := X$ .



**Theorem 31** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a polyhedron. Then, there are finite sets  $V, E \subseteq \mathbb{R}^n$  such that

$$P = \text{conv}(V) + \text{cone}(E).$$

**Proof:** The cone

$$C = \left\{ \begin{pmatrix} x \\ \lambda \end{pmatrix} \mid x \in \mathbb{R}^n, \lambda \in \mathbb{R}, \lambda \geq 0, Ax - \lambda b \leq 0 \right\}$$

is polyhedral, so by the Farkas-Minkowski-Weyl Theorem (Theorem 28), it is generated by finitely many vectors  $\begin{pmatrix} x_1 \\ \lambda_1 \end{pmatrix}, \dots, \begin{pmatrix} x_k \\ \lambda_k \end{pmatrix}$ . Then,  $x \in P$  if and only if  $\begin{pmatrix} x \\ 1 \end{pmatrix} \in C$ , which is the case if and only if  $\begin{pmatrix} x \\ 1 \end{pmatrix} \in \text{cone}\left(\left\{\begin{pmatrix} x_1 \\ \lambda_1 \end{pmatrix}, \dots, \begin{pmatrix} x_k \\ \lambda_k \end{pmatrix}\right\}\right)$ . None of the  $\lambda_i$  can be negative, and by scaling, we can assume  $\lambda_i \in \{0, 1\}$  for  $i = 1, \dots, k$  in any such representation. Then, the sets  $V = \{x_i \mid i \in \{1, \dots, k\}, \lambda_i = 1\}$  and  $E = \{x_i \mid i \in \{1, \dots, k\}, \lambda_i = 0\}$  give a decomposition  $P = \text{conv}(V) + \text{cone}(E)$ .  $\square$

It is easy to check that the Minkowski sum of two polyhedra is again a polyhedron (see exercises). Thus, a set  $P \subseteq \mathbb{R}^n$  is a polyhedron if and only if there are finite sets  $V, E \subseteq \mathbb{R}^n$  such that

$$P = \text{conv}(V) + \text{cone}(E).$$



## 4 Simplex Algorithm

The **SIMPLEX ALGORITHM** by Dantzig [1951] is the oldest algorithm for solving general linear programs. Geometrically it works as follows: Given a polyhedron  $P$  and a linear objective function, we start with any vertex of  $P$ . Then we walk along a one-dimensional face of  $P$  to another vertex and repeat this until we found a vertex where the objective function attains a maximum.

If we want to have a chance to follow this main strategy, we need a pointed polyhedron. That is why in this section we consider linear programs in standard equation form:

$$\begin{aligned} & \max c^t x \\ \text{s.t. } & Ax = b \\ & x \geq 0 \end{aligned} \tag{28}$$

As usual  $A$  is an  $m \times n$ -matrix and  $b$  vector of length  $m$ .

We assume that  $\text{rank}(A) = m$  and that  $Ax = b$  has a feasible solution. These assumptions are no real restrictions because we can run Gaussian elimination on the system  $Ax = b$  in advance (see Section 5.1). Doing this we easily find out if  $Ax = b$  is indeed feasible and we can get rid of redundant constraints, i.e. reduce  $A$  to a set linearly independent rows.

Thus, we also have  $m \leq n$ . If  $m = n$ , then there is only one vector  $x$  with  $Ax = b$ . We can compute this vector (again by using Gaussian elimination) and check if it is non-negative. This solves the linear program in this case. Hence we assume that  $m < n$ .

We are interested in vertices of  $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$ , so in particular, we ask for vectors  $x^* \in \mathbb{R}^n$  with  $Ax^* = b$ ,  $x^* \geq 0$  such that (at least)  $n - m$  entries of  $x^*$  are zero (since  $n$  constraints must be satisfied with equality).

The **SIMPLEX ALGORITHM** works on linear programs in standard equation form (see (28)). Nevertheless, in examples we will often start with LPs in the following form:

$$\begin{aligned} & \max c^t x \\ \text{s.t. } & \tilde{A}x \leq b \\ & x \geq 0 \end{aligned} \tag{29}$$

By adding (non-negative) slack variables  $\tilde{x}$  we get a special case of an LP in standard equation form (with  $A = [\tilde{A} \ I_m]$ ). These LPs of the form  $\max\{c^t x \mid \tilde{A}x + I_m \tilde{x} = b, x \geq 0, \tilde{x} \geq 0\}$  have the advantage that, provided that  $b \geq 0$ , one can easily compute a vertex of the corresponding polyhedron (setting  $x = 0$  and  $\tilde{x}_i = b_i$  for  $i \in \{1, \dots, m\}$  gives a vertex). Such a vertex is needed to start the **SIMPLEX ALGORITHM**.

## 4.1 Feasible Basic Solutions

**Notation:** We denote the index set of the columns of a matrix  $A \in \mathbb{R}^{m \times n}$  by  $\{1, \dots, n\}$ . For a subset  $B \subseteq \{1, \dots, n\}$ , we denote by  $A_B$  the sub-matrix of  $A$  containing exactly the columns with index in  $B$ . Similarly, for a vector  $x \in \mathbb{R}^n$ , we denote by  $x_B$  the sub-vector of  $x$  containing the entries with index in  $B$ . Note that  $x_B$  is a vector of length  $|B|$  but its entries are not indexed from 1 to  $|B|$ , but the indices are the elements of  $B$ , so for example for  $B = \{2, 4, 9\}$  we have  $x_B = (x_2, x_4, x_9)$ .

**Definition 13** Let  $A \in \mathbb{R}^{m \times n}$  be a matrix with rank  $m$  and  $b \in \mathbb{R}^m$  a vector. Let  $B \subseteq \{1, \dots, n\}$  with  $|B| = m$  such that  $A_B$  is regular (i.e. invertible). Set  $N := \{1, \dots, n\} \setminus B$ .

- (a) We call  $B$  a **basis** of  $A$ . The vector  $x$  with  $x_B = A_B^{-1}b$  and  $x_N = 0$  is called **basic solution of  $Ax = b$  for the basis  $B$** .
- (b) If  $x$  is a basic solution of  $Ax = b$  for  $B$ , then the variables  $x_j$  with  $j \in B$  are called **basic variables** and the variables  $x_j$  with  $j \in N$  are called **non-basic variables**.
- (c) A basic solution  $x$  is called **feasible** if  $x \geq 0$ . A basis is called **feasible** if its basic solution is feasible.
- (d) A feasible basic solution  $x$  for a basis  $B$  is called **non-degenerated** if  $A_B^{-1}b > 0$ . Otherwise it is called **degenerated**.

**Remark:** We also use the above definition for inequality systems of the type  $\tilde{A}x \leq b, x \geq 0$  (with  $\tilde{A} \in \mathbb{R}^{m \times \tilde{n}}$ ). E.g. we call a vector  $x^* \in \mathbb{R}^{\tilde{n}}$  with  $\tilde{A}x^* \leq b$  and  $x^* \geq 0$  a basic solution if  $x^*, s^*$  with  $s^* := b - \tilde{A}x^*$  is a basic solution for  $\tilde{A}x + I_m s = b, x \geq 0, s \geq 0$  (with  $n := \tilde{n} + m$  variables). In particular, in a feasible basic solution of  $\tilde{A}x \leq b, x \geq 0$ , the number of tight constraints (including non-negativity constraints) must be at least  $n - m = \tilde{n}$ , and in a *non-degenerated* feasible basic solution, the number of tight constraints must be *exactly*  $\tilde{n}$ . This is because each positive non-slack variable and each positive slack variable is associated with a non-tight constraint.

**Example:** Consider the following system of equations:

$$\begin{array}{rcccccc} x_1 & + & x_2 & + & s_1 & & = & 1 \\ 2x_1 & + & x_2 & & & + & s_2 & = & 2 \\ x_1 & , & x_2 & , & s_1 & , & s_2 & \geq & 0 \end{array} \quad (30)$$

The variables are  $x_1, x_2, s_1$ , and  $s_2$ . We denoted the last two variables by  $s_1$  and  $s_2$  because they can be interpreted as slack variables for the following system of inequalities:  $x_1 + x_2 \leq 1, 2x_1 + x_2 \leq 2, x_1, x_2 \geq 0$ .

If we write the system of equations in matrix notation, we get:

$$\begin{pmatrix} 1 & 1 & 1 & 0 \\ 2 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ s_1 \\ s_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

For  $B = \{1, 2\}$ , we get  $A_B = \begin{pmatrix} 1 & 1 \\ 2 & 1 \end{pmatrix}$  with feasible basis solution  $(1, 0, 0, 0)$ . So in particular this basic feasible solution is degenerated. If we choose instead  $B = \{2, 3\}$ , we get  $A_B = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$  and the corresponding basic solution is  $(0, 2, -1, 0)$  which is, of course, infeasible.

Figure 5 illustrates these two basic solutions. However, note that the figure does not show the solution space (which is 4-dimensional) but only the solution space of the problem without the slack variables  $s_1$  and  $s_2$ , i.e. the solution space of the system  $x_1 + x_2 \leq 1, 2x_1 + x_2 \leq 2, x_1, x_2 \geq 0$ . So the two points  $(1, 0)$  and  $(0, 2)$  are basic solutions only in the sense of the remark stated after the last definition.

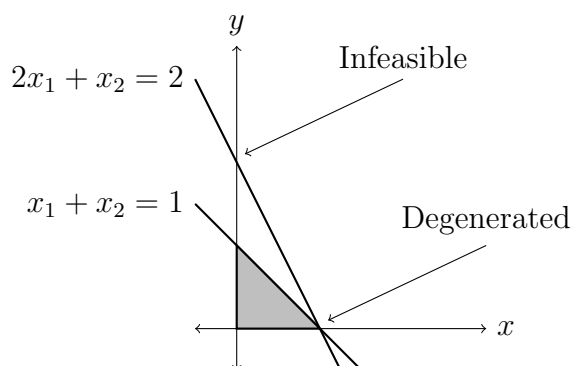


Fig. 5: Infeasible and degenerated basic solutions of (30) to  $\mathbb{R}^2$ .

In this example we could easily make the degenerated basic solution non-degenerated by skipping the redundant constraint  $2x_1 + x_2 \leq 2$ . This is always possible if we only have two non-slackness variables but already in three dimensions there are instances where we cannot get rid of degenerated basic solutions. As an example consider Figure 6. If the pyramid defines the set of all feasible solution the marked vector is a degenerated basic solution, because four constraints are fulfilled with equality while there are only three non-slack variables.

Note that the example (30) shows that the same vertex of a polyhedron can belong to a degenerated or a non-degenerated basic solution, depending on how we describe the polyhedron by a system of inequalities.

**Theorem 32** *Let  $P = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$  be a polyhedron with  $\text{rank}(A) = m < n$ . Then a vector  $x' \in P$  is a vertex of  $P$  if and only if it is a feasible basic solution.*

Degenerated basic solution

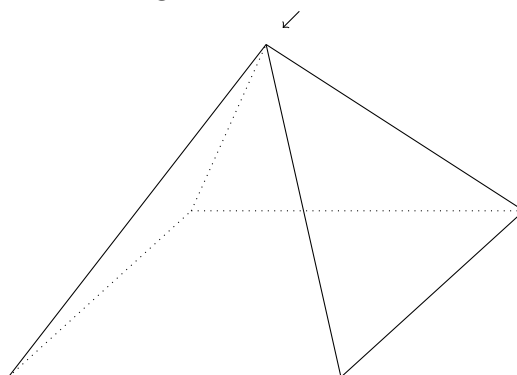


Fig. 6: A degenerated point in  $\mathbb{R}^3$ .

**Proof:** The vector  $x'$  is a vertex of  $P$  if and only if it is a feasible solution of the following system and fulfills  $n$  linearly independent inequalities of the system with equality:

$$\begin{aligned} Ax &\leq b \\ -Ax &\leq -b \\ -I_n x &\leq 0 \end{aligned}$$

This is the case if and only if  $x' \geq 0$ ,  $Ax' = b$  and  $x'_N = 0$  for a set  $N \subseteq \{1, \dots, n\}$  with  $|N| = n - m$  such that with  $B = \{1, \dots, n\} \setminus N$  the matrix  $A_B$  has full rank. This is equivalent to being a feasible basic solution.  $\square$

## 4.2 The Simplex Method

Before we describe the algorithm in general, we will present some examples (which are taken from Matoušek and Gärtner [2007]).

Consider the following linear program:

$$\begin{aligned} \max \quad & x_1 + x_2 \\ \text{s.t.} \quad & -x_1 + x_2 + x_3 = 1 \\ & x_1 + x_4 = 3 \\ & x_2 + x_5 = 2 \\ & x_1, x_2, x_3, x_4, x_5 \geq 0 \end{aligned}$$

$$\begin{pmatrix} -1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 3 \\ 2 \end{pmatrix}$$

We first need a basis to start with. We simply choose  $B = \{3, 4, 5\}$ , which gives us the basic solution  $x = (0, 0, 1, 3, 2)$ . We write the constraints and the objective function in a so-called **simplex tableau**:

$$\begin{array}{rcl} x_3 & = & 1 + x_1 - x_2 \\ x_4 & = & 3 - x_1 \\ x_5 & = & 2 \qquad \qquad - x_2 \\ \hline z & = & \qquad \qquad x_1 + x_2 \end{array}$$

The first three rows describe an equation system that is equivalent to the given one but each basic variable is written as a combination of the non-basic variable. The last line describes the objective function.

We will try to increase non-basic variables (which are zero in the current solution) with a positive coefficient in the objective function. Hence, here we could use  $x_1$  or  $x_2$ , and we choose  $x_2$ . Equation  $x_3 = 1 + x_1 - x_2$  is the critical constraint that prevents us from increasing to something bigger than 1 (without increasing  $x_1$ ). If we set  $x_2$  to something bigger than 1,  $x_3$  would become negative. The constraint  $x_5 = 2 - x_2$  only gives an upper bound of 2 for the value of  $x_2$ . Since the bound induced by non-negativity of  $x_3$  is tighter (so the constraint  $x_3 = 1 + x_1 - x_2$  is critical), we replace 3 in the basis by 2. The new basic variable  $x_2$  can be written as a combination of the non-basic variables by using the first constraint:  $x_2 = 1 + x_1 - x_3$ . The new base is  $B = \{2, 4, 5\}$  with a new basic solution  $x = (0, 1, 0, 3, 1)$ . This is the new simplex tableau:

$$\begin{array}{rcl} x_2 & = & 1 + x_1 - x_3 \\ x_4 & = & 3 - x_1 \\ x_5 & = & 1 - x_1 + x_3 \\ \hline z & = & 1 + 2x_1 - x_3 \end{array}$$

Increase  $x_1$ .  $x_5 = 1 - x_1 + x_3$  is critical.  $x_1 = 1 + x_3 - x_5$ . New base  $B = \{1, 2, 4\}$ .  $x = (1, 2, 0, 2, 0)$ .

$$\begin{array}{rcl} x_1 & = & 1 + x_3 - x_5 \\ x_2 & = & 2 \qquad \qquad - x_5 \\ x_4 & = & 2 - x_3 + x_5 \\ \hline z & = & 3 + x_3 - 2x_5 \end{array}$$

Increase  $x_3$ .  $x_4 = 2 - x_3 + x_5$  is critical.  $x_3 = 2 - x_4 + x_5$ . New base  $B = \{1, 2, 3\}$ .  $x = (3, 2, 2, 0, 0)$ .

$$\begin{array}{rcl} x_1 & = & 3 - x_4 \\ x_2 & = & 2 \qquad \qquad - x_5 \\ x_3 & = & 2 - x_4 + x_5 \\ \hline z & = & 5 - x_4 - x_5 \end{array}$$

The value of the objective function for any feasible solution  $(x_1, \dots, x_5)$  is  $5 - x_4 - x_5$ . Since we have found a solution where  $x_4 = x_5 = 0$  and we have the constraint that  $x_i \geq 0$  ( $i = 1, \dots, 5$ ),

our solution is an optimum solution.

### Unbounded instance:

As a second example, consider:

$$\begin{array}{rllll} \max & x_1 & & & \\ \text{s.t.} & x_1 - x_2 + x_3 & & & = 1 \\ & -x_1 + x_2 & & + x_4 & = 2 \\ & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq 0 \end{array}$$

Quite obviously this LP is unbounded (one can choose  $x_1$  arbitrarily large and set  $x_2 = x_1$ ,  $x_3 = 1$ , and  $x_4 = 2$ ).

Again we use the “slack variables” (here  $x_3$  and  $x_4$ ) for a first basis. This gives  $B = \{3, 4\}$  and  $x = (0, 0, 1, 2)$ .

$$\begin{array}{r} x_3 = 1 - x_1 + x_2 \\ x_4 = 2 + x_1 - x_2 \\ \hline z = x_1 \end{array}$$

Increase  $x_1$ .  $x_3 = 1 - x_1 + x_2$  is critical.  $x_1 = 1 + x_2 - x_3$ . New base  $B = \{1, 4\}$ .  $x = (1, 0, 0, 3)$ .

$$\begin{array}{r} x_1 = 1 + x_2 - x_3 \\ x_4 = 3 - x_3 \\ \hline z = 1 + x_2 - x_3 \end{array}$$

We can increase  $x_2$  as much as we want (provided that we increase  $x_1$  by the same amount). Thus the simplex tableau show that the linear program is unbounded.

### Degeneracy:

A final example shows what may happen if we get a degenerated basic solution.

$$\begin{array}{rllll} \max & & x_2 & & \\ \text{s.t.} & -x_1 + x_2 + x_3 & & & = 0 \\ & x_1 & & + x_4 & = 2 \\ & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq 0 \end{array}$$

Starting basis is  $B = \{3, 4\}$ , so  $x = (0, 0, 0, 2)$  which is a degenerated solution.

$$\begin{array}{r} x_3 = x_1 - x_2 \\ x_4 = 2 - x_1 \\ \hline z = x_2 \end{array}$$



We want to increase  $x_2$ .  $x_3 = x_1 - x_2$  is critical.  $x_2 = x_1 - x_3$ . We will replace 3 by 2 in the basis. However, we cannot increase  $x_2$ . New base  $B = \{2, 4\}$ .  $x = (0, 0, 0, 2)$ .

$$\begin{array}{rcl} x_2 & = & x_1 - x_3 \\ x_4 & = & 2 - x_1 \\ \hline z & = & x_1 - x_3 \end{array}$$

Increase  $x_1$ .  $x_4 = 2 - x_1$  is critical.  $x_1 = 2 - x_4$ . New base  $B = \{1, 2\}$ .  $x = (2, 2, 0, 0)$ .

$$\begin{array}{rcl} x_1 & = & 2 - x_4 \\ x_2 & = & 2 - x_3 - x_4 \\ \hline z & = & 2 - x_3 - x_4 \end{array}$$

Again, we have found an optimum solution because all coefficients of the non-basic variables in the objective function  $z = 2 - x_3 - x_4$  are negative.

After these three examples, we will now describe the simplex method in general.

For a basis  $B$ , the **simplex tableau** is a system  $T(B)$  of  $m + 1$  linear equations with variables  $x_1, \dots, x_n$  and  $z$  with this form

$$\begin{array}{rcl} x_B & = & p + Qx_N \\ z & = & z_0 + r^t x_N \end{array} \quad (31)$$

and the following properties:

- $x_B$  is the vector of the basic variables,  $N = \{1, \dots, n\} \setminus B$ , and  $x_N$  is the vector of the non-basic variables,
- $T(B)$  has the same set of solutions as the system  $Ax = b$ ,  $z = c^t x$ .
- $p$  is a vector of length  $m$ ,  $Q$  is an  $m \times (n - m)$ -matrix,  $r$  is a vector of length  $n - m$ , and  $z_0 \in \mathbb{R}$ .

Note that the entries of  $p$  are not necessarily numbered from 1 to  $m$  but that  $p$  uses  $B$  as the set of indices (and for  $r$ , we have a corresponding statement). In particular, the rows of  $Q$  are indexed by  $B$  and the columns by  $N$ . We denote the entries of  $Q$  by  $q_{ij}$  (where  $i \in B$  and  $j \in N$ ).

**Lemma 33** *For each basis  $B$ , there is a simplex tableau  $T(B)$ .*

**Proof:** Set  $p = A_B^{-1}b$ ,  $Q = -A_B^{-1}A_N$ ,  $r = c_N - (c_B^t A_B^{-1} A_N)^t$ , and  $z_0 = c_B^t A_B^{-1}b$ .

Then  $x_B = A_B^{-1}b - A_B^{-1}A_N x_N$  which is equivalent to  $A_B x_B = b - A_N x_N$  and  $Ax = b$ .

Moreover,  $z = c_B^t A_B^{-1} b + (c_N^t - (c_B^t A_B^{-1} A_N)) x_N = c_B^t A_B^{-1} (b - A_N x_N) + c_N^t x_N = c_B^t A_B^{-1} A_B x_B + c_N^t x_N = c_B^t x_B + c_N^t x_N = c^t x$  for any solution  $x$  of  $Ax = b$ .  $\square$

**Remark:** It is easy to check that there is only *one* simplex tableau for every feasible basis  $B$ .

The cost function  $z_0 + r^t x_N$  does not directly depend on the basic variables but only on the non-basic variables. Their impact on the overall cost is given by the vector  $r = c_N - (c_B^t A_B^{-1} A_N)^t$ . An entry of  $r$  is called the **reduced cost** of its corresponding non-basic variable.

If all reduced costs are non-positive, we have already found an optimum solution:

**Lemma 34** *Let  $T(B)$  be a simplex tableau for a feasible basis  $B$ . If  $r \leq 0$ , then the basic solution of  $B$  is optimum.*

**Proof:** Let  $x$  be the basic solution of  $B$ . Since  $x_N = 0$ , we have  $c^t x = z_0 (= c_B^t A_B^{-1} b)$ . If  $x^*$  is any feasible solution with value  $z^* = c^t x^*$ , then  $x^*$  and  $z^*$  are also a solution of  $T(B)$ , and we have (because of  $r \leq 0$  and  $x_N^* \geq 0$ )  $z^* = z_0 + r^t x_N^* \leq z_0 = c^t x$ .  $\square$

**Lemma 35** *Let  $T(B)$  be a simplex tableau for a feasible basis  $B$ . If there is an  $\alpha \in N$  with  $r_\alpha > 0$  such that the column of  $Q$  with index  $\alpha$  contains non-negative entries only, the linear program is unbounded.*

**Proof:** Let  $x$  the feasible basic solution for  $B$ . Let  $K \in \mathbb{R}$  with  $K > c^t x$  be a constant. Define a new feasible solution  $\tilde{x}$  as follows:  $\tilde{x}_\alpha := \frac{K - c^t x}{r_\alpha}$ ,  $\tilde{x}_i = x_i$  for  $i \in N \setminus \{\alpha\}$ , and  $\tilde{x}_j := p_j + q_{j\alpha} \tilde{x}_\alpha$  for  $j \in B$ . It is easy to check that  $\tilde{x}$  is a feasible solution with  $c^t \tilde{x} \geq K$ . Hence, the linear program is unbounded.  $\square$

In the following, we denote the entries of  $A$  by  $a_{ij}$  ( $i \in \{1, \dots, m\}$ ,  $j \in \{1, \dots, n\}$ ). The column of  $A$  with index  $j$  is denoted by  $a_j$ .

**Lemma 36** *Let  $T(B)$  be a simplex tableau for a feasible basis  $B$ . Let  $\alpha \in N$  be an index with  $r_\alpha > 0$  and  $\beta \in B$  with  $q_{\beta\alpha} < 0$  and  $\frac{p_\beta}{q_{\beta\alpha}} = \max\{\frac{p_i}{q_{i\alpha}} \mid q_{i\alpha} < 0, i \in B\}$ . Then  $\tilde{B} = (B \cup \{\alpha\}) \setminus \{\beta\}$  is a feasible basis.*

**Proof:** We have to show that  $A_{\tilde{B}}$  has full rank and that it is feasible i.e. that its basic solution is non-negative.

(i)  $\tilde{B}$  is a basis: We will show that  $A_{\tilde{B}}^{-1} A_{\tilde{B}}$  has full rank.

All but one columns of  $A_{\tilde{B}}$  belong to  $A_B$ . Hence, the matrix  $A_{\tilde{B}}^{-1} A_{\tilde{B}}$  contains all unit vectors  $e_i$  with the possible exception of  $e_\beta$  because we removed the  $\beta$ -th column from  $A_B$ . However, this removed column has been replaced by the  $\alpha$ -th column  $a_\alpha$  of  $A$ , so

the remaining column of  $A_B^{-1}A_{\tilde{B}}$  is  $A_B^{-1}a_{\cdot\alpha}$ . But this is exactly the column with index  $\alpha$  of  $-Q = A_B^{-1}A_N$ . By construction,  $q_{\beta\alpha} \neq 0$ , so all columns of  $A_B^{-1}A_{\tilde{B}}$  are linearly independent.

- (ii) We have to show that the basic solution of  $\tilde{B}$  is non-negative. We increase  $x_\alpha$  to  $-\frac{p_\beta}{q_{\beta\alpha}}$  and set the basic variables  $x_B$  to  $p - q_{\cdot\alpha}\frac{p_\beta}{q_{\beta\alpha}}$ , where  $q_{\cdot\alpha}$  is the column with index  $\alpha$  of  $Q$ . For  $i \in B$  with  $q_{i\alpha} \geq 0$  (so in particular  $i \neq \beta$ ) we have  $p_i - q_{i\alpha}\frac{p_\beta}{q_{\beta\alpha}} \geq p_i \geq 0$ . For  $i \in B$  with  $q_{i\alpha} < 0$  we have  $\frac{p_\beta}{q_{\beta\alpha}} \geq \frac{p_i}{q_{i\alpha}}$ , so  $p_i \geq q_{i\alpha}\frac{p_\beta}{q_{\beta\alpha}}$  with equality in the last inequality for  $i = \beta$ . This leads to  $x_\beta = 0$  and  $x_B \geq 0$ , so we get a feasible basic solution for  $\tilde{B}$ .  $\square$

---

**Algorithm 1:** Simplex Algorithm

---

**Input:** A matrix  $A \in \mathbb{R}^{m \times n}$ , a vector  $b \in \mathbb{R}^m$ , and a vector  $c \in \mathbb{R}^n$

**Output:** A vector  $\tilde{x} \in \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$  maximizing  $c^t x$  or the message that  $\max\{c^t x \mid Ax = b, x \geq 0\}$  is unbounded or infeasible

- 1 Compute a feasible basis  $B$ ;
- 2 If no such basis exists, stop with the message “INFEASIBLE”;
- 3 Set  $N = \{1, \dots, n\} \setminus B$  and compute the feasible basic solution  $x$  for  $B$ ;  
//  $x_B = A_B^{-1}b, x_N = 0$ .
- 4 Compute the simplex tableau T(B)

$$\begin{array}{rcl} x_B & = & p + Qx_N \\ z & = & z_0 + r^t x_N \end{array}$$

for the basis  $B$ ; // See equation (31) and the following notation.

- 5 **if**  $r \leq 0$  **then**  
└ **return**  $\tilde{x} = x$ ; //  $\tilde{x}$  is optimum (see Lemma 34).
  - 6 Choose an index  $\alpha \in N$  with  $r_\alpha > 0$ ;  
// Here we can apply different pivot rules.
  - 7 **if**  $q_{i\alpha} \geq 0$  for all  $i \in B$  **then**  
└ **return** “UNBOUNDED”; // By Lemma 35, the LP is unbounded.
  - 8 Choose an index  $\beta \in B$  with  $q_{\beta\alpha} < 0$  and  $\frac{p_\beta}{q_{\beta\alpha}} = \max\{\frac{p_i}{q_{i\alpha}} \mid q_{i\alpha} < 0, i \in B\}$ ;  
// Again, we can apply different pivot rules.
  - 9 Set  $B = (B \setminus \{\beta\}) \cup \{\alpha\}$ ;  
// See Lemma 36 proving that we get a new feasible basis.
  - 10 **go to** line 3
- 

Algorithm 1 summarizes the **SIMPLEX ALGORITHM**.

**Remark:** In line 1 of the algorithm we have to compute an initial feasible basis. This can be done with the following trick: We assume that the **SIMPLEX ALGORITHM** works correctly and has a finite running time, provided that we can compute an initial basis. We further assume that  $b \geq 0$  (otherwise, we have to multiply some equations by -1 first). Then, we set  $\tilde{A} = (A \mid I_m)$ , add new variables  $x_{n+1}, \dots, x_{n+m}$ , and solve (with  $\tilde{x} = (x_1, \dots, x_{n+m})$ ) the following problem:

$$\begin{aligned}
\max \quad & -(x_{n+1} + x_{n+2} + \dots + x_{n+m}) \\
\text{s.t.} \quad & \tilde{A}\tilde{x} = b \\
& \tilde{x} \geq 0
\end{aligned} \tag{32}$$

For this linear program, it is trivial to find a feasible basis ( $\{n+1, \dots, n+m\}$  will work), so we can solve it by the **SIMPLEX ALGORITHM**. If the value of its optimum solution is negative, this means that the original linear program does not have a feasible solution. Otherwise, the **SIMPLEX ALGORITHM** will provide a basic solution for the original linear program. In this case, the solution of the new LP computed by the **SIMPLEX ALGORITHM** could contain variables from  $x_{n+1}, \dots, x_{n+m}$  as basic variables but their value must be 0 and hence they can be replaced easily by variables from  $x_1, \dots, x_n$ .

In lines 6 and 8, we may have a choice between different candidates to enter or leave the basis. The elements chosen in these steps are called **pivot elements**, and the rules by which we choose them are called **pivot rules**. Several different pivot rules for the entering variable have been proposed:

- **Largest coefficient rule:** For the entering variable choose  $\alpha$  such that  $r_\alpha$  is maximized. This is the rule that was proposed by Dantzig in his first description of the **SIMPLEX ALGORITHM**.
- **Largest increase rule:** Choose the entering variable such that the increase of the objective function is maximized. Finding an  $\alpha$  with that property takes more time because it is not sufficient to consider the vector  $r$  only.
- **Steepest edge rule:** Choose the entering variable in such a way that we move the feasible basic solution in a direction as close to the direction of the vector  $c$  as possible. This means we maximize

$$\frac{c^t(x_{new} - x_{old})}{\|x_{new} - x_{old}\|}$$

where  $x_{old}$  is the basic feasible solution of the current basis and  $x_{new}$  is the basic feasible solution of the basis after the exchange step. This rule is even more timing-consuming but in many practical experiments it turned out to lead to a small number of exchange steps.

Here, we only analyze a pivot rule that is quite inefficient in practice but has the nice property that we can show that the **SIMPLEX ALGORITHM** terminates at all, if we follow that rule. If all exchange steps improve the value of the current solution, we can be sure that the algorithm will terminate because we can never visit the same basic solution twice, and there is only a finite (though exponential) number of basic solutions. However, exchange steps do not necessarily change the value of the solution. Therefore, depending on the pivot rules, it is possible that the **SIMPLEX ALGORITHM** runs in an endless loop by considering the same sequence of bases forever. This behavior is called **cycling** (see page 30 ff. of Chvátal [1983] for an example that this can really happen). The good news is that we can avoid cycling by using an appropriate pivot rule.

If the algorithm does not terminate, it has to consider the same basis  $B$  twice. The computation between two occurrences of  $B$  is called a cycle. Let  $F \subseteq \{1, \dots, n\}$  be the indices of the variables that have been added to (and hence removed from) the basis during one cycle. We call  $x_F$  the cycle variables.

**Lemma 37** *If the SIMPLEX ALGORITHM cycles, all basic solutions during the cycling are the same and all cycle variables are 0.*

**Proof:** The value of a solution considered in SIMPLEX ALGORITHM never decreases, so during cycling it cannot increase either. Let  $B$  be a feasible basis that occurs in the cycle, and let  $B' = (B \cup \{\alpha\}) \setminus \{\beta\}$  be the next basis. The only non-basic variable that could be increased is  $x_\alpha$ . However, if it indeed was increased, then, because  $r_\alpha > 0$ , this would increase the value of the solution. This shows that the non-basic variables remain zero. But then, all variables remain unchanged because the basic variables are determined uniquely by the non-basic variables.  $\square$

A pivot rule that is able to avoid cycling is **Bland's rule** (Bland [1977]) that can be described as follows: In line 6 of the SIMPLEX ALGORITHM, we choose  $\alpha$  among all elements in  $N$  with  $r_\alpha > 0$  such that  $\alpha$  is minimal. In line 8, we choose  $\beta$  among all elements in  $B$  with  $q_{\beta\alpha} < 0$  and  $\frac{p_\beta}{q_{\beta\alpha}} = \max\{\frac{p_i}{q_{i\alpha}} \mid q_{i\alpha} < 0, i \in B\}$  such that  $\beta$  is minimal.

**Theorem 38** *With Bland's rule as pivot rule in lines 6 and 8, the SIMPLEX ALGORITHM terminates after a finite number of steps.*

**Proof:** Assume that the algorithm cycles while using Bland's rule. We use the notation from above and consider the set  $F$  of the indices of the cycle variables. Let  $\pi$  be the largest element of  $F$ , and let  $B$  be the basis just before  $\pi$  enters the basis. Let  $p, Q, r$  and  $z_0$  be the entries of the simplex tableau  $T(B)$ . Let  $B'$  be the basis just before  $\pi$  leaves it. Let  $p', Q', r'$  and  $z'_0$  be the entries of the simplex tableau  $T(B')$ .

Let  $N = \{1, \dots, n\} \setminus B$  be the set of the non-basic variables (so in particular  $\pi \in N$ ). According to Bland's rule we choose the smallest index and  $\pi = \max(F)$ , so when  $B$  is considered,  $\pi$  is the only candidate in  $F$  to enter the basis. In other words:

$$r_\pi > 0 \text{ and } r_j \leq 0 \text{ for all } j \in N \cap (F \setminus \{\pi\}). \quad (33)$$

Let  $\alpha$  be the index entering  $B'$ . Again by Bland's rule,  $\pi$  must have been the only candidate among all elements of  $F$  to leave  $B'$ . Since  $p'_j = 0$  for all  $j \in B' \cap F$ , this means that

$$q'_{\pi\alpha} < 0 \text{ and } q'_{j\alpha} \geq 0 \text{ for } j \in B' \cap (F \setminus \{\pi\}). \quad (34)$$

Roughly spoken, we will get a contradiction because (33) says that in a feasible basic solution increasing a non-basic variable in  $x_{F \setminus \{\pi\}}$  or decreasing  $x_\pi$  (to something negative!) will not

improve the result. On the other hand, (34) says that increasing  $x_\alpha$  while decreasing  $x_\pi$  (again to something negative) will improve the result.

We will formalize this statement by considering the following auxiliary linear program:

$$\begin{aligned}
\max \quad & c^t x \\
\text{s.t.} \quad & Ax = b \\
& x_{F \setminus \{\pi\}} \geq 0 \\
& x_\pi \leq 0 \\
& x_{N \setminus F} = 0
\end{aligned} \tag{35}$$

Note that there are no constraints on the signs of the variables in  $x_{B \setminus F}$ .

We will show two claims that obviously cause a contradiction:

Claim 1: The LP (35) has an optimum solution.

Proof of Claim 1: Let  $\tilde{x}$  be a basic feasible solution (of the original LP) of the basis  $B$ . We have  $\tilde{x}_F = 0$ , so in particular  $\tilde{x}_\pi = 0$ , and hence  $\tilde{x}$  is a feasible solution of (35). The cost of any solution  $x$  of  $Ax = b$  can be written as  $c^t x = z_0 + r^t x_N$ . For any solution  $x$  of (35), we have

$$x_j \begin{cases} \geq 0 & \text{if } j \in F \setminus \{\pi\} \\ \leq 0 & \text{if } j = \pi \end{cases}$$

Therefore, by statement (33),  $r_j x_j \leq 0$  for all  $j \in n \cap F$ . With the condition  $x_{N \setminus F} = 0$  this leads to  $r^t x_N \leq 0$  for any solution  $x$  of (35). Therefore, the value of any such solution is at most  $z_0$ , and thus  $\tilde{x}$  is an optimum solution of (35). This proves Claim 1.

Claim 2: The LP (35) is unbounded.

The bases are changed during the cycling but we always have the same basic solution. Hence, if  $\tilde{x}$  is a feasible basic solution of the original LP for basis  $B$  is also a feasible basic solution for the basis  $B'$ . We choose a positive number  $K$  and set  $x'_\alpha = K$ . For  $j \in N' \setminus \{\alpha\}$  (with  $N' = \{1, \dots, n\} \setminus B'$ ), we set  $x'_j = \tilde{x}_j = 0$ . Moreover, we set  $x_{B'} = p' + Q' x'_{N'}$ . By (34), this defines a feasible solution of the auxiliary LP (35). Since  $\alpha$  was a candidate for entering the basis  $B'$ , we have  $r'_\alpha > 0$ . Hence, we get a solution with value  $c^t x' = z'_0 + r'^t x'_{N'} = z'_0 + K \cdot r'_\alpha$ . As we can choose  $K$  arbitrarily large, this shows that LP (35) is unbounded.  $\square$

### 4.3 Efficiency of the Simplex Algorithm

We have seen that Bland's rule guarantees that the `SIMPLEX ALGORITHM` will terminate. What can we say about the running time? Consider for some  $\epsilon$  with  $0 < \epsilon < \frac{1}{2}$  the following example:

$$\begin{aligned} \max x_n \\ -x_1 &\leq 0 \\ x_1 &\leq 1 \\ \epsilon x_{j-1} - x_j &\leq 0 \quad \text{for } j \in \{2, \dots, n\} \\ \epsilon x_{j-1} + x_j &\leq 1 \quad \text{for } j \in \{2, \dots, n\} \end{aligned}$$

Of course, adding non-negativity constraints for all variables would not change the problem. The polyhedron defined by these inequalities is called **Klee-Minty cube** (Klee and Minty [1972]). It turns out that the `SIMPLEX ALGORITHM` with Bland's rule (depending on the initial solution) may consider  $2^n$  bases before finding the optimum solution. In particular, this example shows that we don't get a polynomial-time algorithm.

The bad news is that for any of the above pivot rules instances have been found where the `SIMPLEX ALGORITHM` with that particular pivot rule has exponential running time.

Assume that you are given an optimum pivot rule that guides you to an optimum solution with a smallest possible number of iterations. Then, the number of iterations depends on the following property of the instances:

**Definition 14** *The combinatorial diameter of a pointed polyhedron  $P$  is the diameter (i.e. the largest distance of two nodes) of the undirected graph  $G_P$ , where  $V(G_P)$  is the set of vertices of  $P$  and two nodes  $v, w \in V(G_P)$  are connected by an edge in  $G_P$  if and only if there is a face of dimension 1 containing  $v$  and  $w$ .*

Obviously, if we don't make any assumptions on the starting solution, the number of iterations performed by the `SIMPLEX ALGORITHM` optimizing over a polyhedron  $P$  will be at least the combinatorial diameter of  $P$ , even with an optimum pivot rule.

It is an open question what the largest combinatorial diameter of a  $d$ -dimensional polyhedron with  $n$  facets is. In 1957, W. Hirsch conjectured that the combinatorial diameter could be at most  $n - d$ . This conjecture was open for decades but it has been disproved by Santos [2011] who showed that there is a 20-dimensional polyhedron with 40 facets and combinatorial diameter 21. More generally, he proved that there are counter-examples to the Hirsch conjecture with arbitrarily many facets. Nevertheless, it is still possible that the combinatorial diameter is always polynomially (or even linearly) bounded in the dimension and the number of facets. The best known upper bound for the combinatorial diameter is  $O(n^{2+\log d})$  and was proven by Kalai and Kleitman [1992]. For an overview of this topic see Section 3.3 of Ziegler [2007].

In practical experiments, the **SIMPLEX ALGORITHM** typically turns out to be very efficient. It could also be proved that the average running time (with a specified probabilistic model) is polynomial (see Borgwardt [1982]). Moreover, Spielmann and Teng [2005] have shown that the expected running time on a slight perturbation of a worst-case instance can be bounded by a polynomial (“smoothed analysis”, see also Dadush and Hulberts [2019])

### Revised Simplex Algorithm

If one implements the **SIMPLEX ALGORITHM** as described above, an explicit computation of the simplex tableau can be time-consuming. This can be avoided in the so-called **REVISED SIMPLEX ALGORITHM**. In particular, we do not have to store the  $m \times (n - m)$ -matrix  $Q$  completely. It is sufficient to compute the column of  $Q$  with index  $\alpha$  after we have found an  $\alpha \in N$  with  $r_\alpha > 0$ . This method is called **column generation**. Moreover, we do not really need the matrix  $A_B^{-1}$ . In fact, we only want to solve equation system of the type  $A_B y = d$ . It is more efficient to compute an LU-decomposition of  $A_B$  and update it after each exchange step.

## 4.4 Dual Simplex Algorithm

If the linear program  $\max\{c^t x \mid Ax = b, x \geq 0\}$  is feasible and bounded then the **SIMPLEX ALGORITHM** does not only provide an optimum primal solution but we can also get an optimum solution of the dual linear program  $\min\{b^t y \mid A^t y \geq c\}$ . To see this, let  $B$  the feasible basis corresponding to the optimum computed by the **SIMPLEX ALGORITHM**. Set  $\tilde{y} = A_B^{-t} c_B$  (where  $A_B^{-t} = (A_B^t)^{-1}$ ). This leads to  $A_B^t \tilde{y} = c_B$  and  $A_N^t \tilde{y} = A_N^t A_B^{-t} c_B \geq c_N$  where the last inequality follows from the fact that in  $T(B)$  we have  $0 \geq r = c_N - (c_B^t A_B^{-1} A_N)^t$ . So the vector  $\tilde{y}$  is feasible for the dual LP, and it is an optimum solution because together with the (primal) basic solution  $\tilde{x}$  for the basis  $B$ , it satisfies the complementary slackness condition  $(\tilde{y}^t A - c^t) \tilde{x} = 0$ .

In fact, the condition  $r \leq 0$  in the simplex tableau  $T(B)$  guarantees the existence of a dual solution  $y$  with  $y^t A_B = c_B^t$ . In the **DUAL SIMPLEX ALGORITHM**, we start with a feasible basic dual solution, i.e. a feasible dual solution for which a basis  $B$  exists with  $y^t A_B = c_B^t$ . If  $c_B^t A_B^{-1}$  is a feasible dual solution, we call  $B$  a *dual feasible basis*. Then, we compute the corresponding simplex tableau  $T(B)$  (which exists for any basis not just a feasible basis). Thus the vector  $r$  will have no positive entry. Note that  $B$  may not be feasible, so entries of  $p$  can be negative. Now the algorithm swaps elements between the basis and the rest of the variables similarly to the simplex algorithm but instead of keeping  $p$  non-negative it keeps  $r$  non-positive.

For any basis  $B$  such that in  $T(B)$  the vector  $r$  has no positive entry, the following properties (that are easy to prove) are the basis of the **DUAL SIMPLEX ALGORITHM**:

- There is a feasible dual solution  $y$  with  $y^t A_B = c_B$ .
- If  $p \geq 0$  then the current dual solution is optimum.
- $z_0$  is the current solution value of the dual solution.



- If there is a  $\beta \in B$  with  $p_\beta < 0$  such that  $q_{\beta j} \leq 0$  for all  $j \in N$ , then the primal LP is infeasible.
- For  $\beta \in B$  with  $p_\beta < 0$  and  $\alpha \in N$  with  $q_{\beta\alpha} > 0$  with  $\frac{r_\alpha}{q_{\beta\alpha}} \geq \frac{r_j}{q_{\beta j}}$  for all  $j \in N$  with  $q_{\beta j} > 0$ , then  $(B \setminus \{\beta\}) \cup \{\alpha\}$  is a dual feasible basis. Then the value of the dual solution is changed by  $\frac{-p_\beta}{q_{\beta\alpha}} r_\alpha$ . In particular, if  $r_\alpha \neq 0$  then the value of the dual solution gets smaller.

The DUAL SIMPLEX ALGORITHM simply applies the exchange steps in the last item until we get a feasible basis. The algorithm can be considered as the SIMPLEX ALGORITHM applied to the dual LP. Thus it can also run into cycling and its efficiency is not better than the efficiency of the SIMPLEX ALGORITHM.

However, in some applications, the DUAL SIMPLEX ALGORITHM is very useful: If you add an additional constraint to the primal LP, then a primal solution can become infeasible, so in the PRIMAL SIMPLEX ALGORITHM we have to start from scratch. However, the dual solution is still feasible. It is possibly not optimal but often it can be made optimal with just some iterations of the DUAL SIMPLEX ALGORITHM.



## 5 Sizes of Solutions

Before we will describe polynomial-time algorithms for solving linear programs we have to make sure that we can store the output and all intermediate results with numbers whose sizes are polynomial in the input size. To this end we have to define the size of numbers. Assuming that all numbers are given in a binary representation, we define for

- $n \in \mathbb{Z}$ :  $\text{size}(n) := 1 + \lceil \log(|n| + 1) \rceil$ ,
- $r = \frac{p}{q}$  with  $p, q \in \mathbb{Z}$ , relatively prime:  $\text{size}(r) := \text{size}(p) + \text{size}(q)$ ,
- vectors  $x = (x_1, \dots, x_n) \in \mathbb{Q}^n$ :  $\text{size}(x) := n + \sum_{i=1}^n \text{size}(x_i)$ ,
- matrices  $A = (a_{ij})_{\substack{i=1, \dots, m \\ j=1, \dots, n}} \in \mathbb{Q}^{m \times n}$ :  $\text{size}(A) := nm + \sum_{i=1}^m \sum_{j=1}^n \text{size}(a_{ij})$ .

**Remark:** In order to get a description of a fraction  $r$  of with  $\text{size}(r)$  bits, we have to write  $r$  as  $\frac{p}{q}$  for numbers  $p, q \in \mathbb{Z}$  that are relatively prime. Therefore, in any computation, when a fraction  $\frac{p}{q}$  arises, we apply the Euclidean Algorithm to  $p$  and  $q$  and divide  $p$  and  $q$  by their greatest common divisor. The Euclidean Algorithm has polynomial running time, so during any algorithm, we can assume that any fraction  $r$  is stored by using just  $\text{size}(r)$  bits.

**Proposition 39** For  $r_1, \dots, r_n \in \mathbb{Q}$ , we have

$$(a) \quad \text{size} \left( \prod_{i=1}^n r_i \right) \leq \sum_{i=1}^n \text{size}(r_i)$$

$$(b) \quad \text{size} \left( \sum_{i=1}^n r_i \right) \leq 2 \sum_{i=1}^n \text{size}(r_i)$$

**Proof:** Both statements are obvious if the numbers  $r_1, \dots, r_n$  are integers. Hence assume that  $r_i = \frac{p_i}{q_i}$  for non-zero numbers  $p_i$  and  $q_i$  that are relatively prime ( $i = 1, \dots, n$ ).

$$(a) \quad \text{size} \left( \prod_{i=1}^n r_i \right) \leq \text{size} \left( \prod_{i=1}^n p_i \right) + \text{size} \left( \prod_{i=1}^n q_i \right) \leq \sum_{i=1}^n \text{size}(p_i) + \sum_{i=1}^n \text{size}(q_i) = \sum_{i=1}^n \text{size}(r_i).$$

$$(b) \quad \text{We have } \text{size} \left( \prod_{i=1}^n q_i \right) \leq \sum_{i=1}^n \text{size}(q_i) \leq \sum_{i=1}^n \text{size}(r_i), \text{ and } \text{size} \left( \sum_{i=1}^n p_i \prod_{j \in \{1, \dots, n\} \setminus \{i\}} q_j \right) \leq \text{size} \left( \prod_{i=1}^n |p_i| \prod_{j=1}^n q_j \right) \leq \sum_{i=1}^n \text{size}(r_i). \text{ Since } \sum_{i=1}^n r_i = \frac{1}{\prod_{i=1}^n q_i} \sum_{i=1}^n p_i \prod_{j \in \{1, \dots, n\} \setminus \{i\}} q_j, \text{ this proves the claim. } \quad \square$$

**Proposition 40** For  $x, y \in \mathbb{Q}^n$ , we have

(a)  $\text{size}(x + y) \leq 2(\text{size}(x) + \text{size}(y))$

(b)  $\text{size}(x^t y) \leq 2(\text{size}(x) + \text{size}(y))$

**Proof:**

(a) We have

$$\text{size}(x + y) = n + \sum_{i=1}^n \text{size}(x_i + y_i) \leq n + 2 \sum_{i=1}^n \text{size}(x_i) + 2 \sum_{i=1}^n \text{size}(y_i) = 2(\text{size}(x) + \text{size}(y)) - 3n.$$

(b) We have

$$\begin{aligned} \text{size}(x^t y) &= \text{size} \left( \sum_{i=1}^n x_i y_i \right) \leq 2 \sum_{i=1}^n \text{size}(x_i y_i) \leq 2 \left( \sum_{i=1}^n \text{size}(x_i) + \sum_{i=1}^n \text{size}(y_i) \right) \\ &= 2(\text{size}(x) + \text{size}(y)) - 4n. \end{aligned}$$

□

**Proposition 41** For any matrix  $A \in \mathbb{Q}^{n \times n}$ , we have  $\text{size}(\det(A)) \leq 2\text{size}(A)$ .

**Proof:** Write the entries  $a_{ij}$  of  $A$  as  $a_{ij} = \frac{p_{ij}}{q_{ij}}$  where  $p_{ij}$  and  $q_{ij}$  are relatively prime ( $i, j = 1, \dots, n$ ). Let  $\det(A) = \frac{p}{q}$  where  $p$  and  $q$  are relatively prime, too.

Then  $|\det(A)| \leq \prod_{i=1}^n \prod_{j=1}^n (|p_{ij}| + 1)$  and  $|q| \leq \prod_{i=1}^n \prod_{j=1}^n |q_{ij}|$ . Therefore,

$$\text{size}(q) \leq \text{size}(A)$$

and  $|p| = |\det(A)||q| \leq \prod_{i=1}^n \prod_{j=1}^n ( (|p_{ij}| + 1)|q_{ij}| )$ . We can conclude

$$\text{size}(p) \leq \sum_{i=1}^n \sum_{j=1}^n (\text{size}(p_{ij}) + 1 + \text{size}(q_{ij})) = \text{size}(A).$$

This proves  $\text{size}(\det(A)) \leq 2\text{size}(A)$ .

□

**Proposition 42** Let  $\max\{c^t x \mid Ax \leq b\}$  be a feasible bounded linear program with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . Then, there is an optimum (rational) solution  $x$  with  $\text{size}(x) \leq 4n(\text{size}(A) + \text{size}(b))$ . If  $b = e_i$  oder  $b = -e_i$  for a unit vector  $e_i$ , then there is a non-singular submatrix  $A'$  of  $A$  and an optimum solution  $x$  with  $\text{size}(x) \leq 4n\text{size}(A')$ .

**Proof:** By Corollary 19 the maximum of  $c^t x$  over  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  must be attained in a minimal face of  $P$ . Let  $F$  be a minimal face where the maximum is attained. By Proposition 22, we can write  $F = \{x \in \mathbb{R}^n \mid \tilde{A}x = \tilde{b}\}$  for some subsystem  $\tilde{A}x \leq \tilde{b}$  of  $Ax \leq b$ . We can assume that the rows of  $\tilde{A}$  are linearly independent. Choose  $B \subseteq \{1, \dots, n\}$  such that  $\tilde{A}_B$  is a regular square matrix. Then  $x \in \mathbb{R}^n$  with  $x_B = \tilde{A}_B^{-1} \tilde{b}$  and  $x_N = 0$  (with  $N = \{1, \dots, n\} \setminus B$ ) is an optimum solution of the linear program. By Cramer's rule the entries of  $x_B$  can be written as  $x_j = \frac{\det(\tilde{A}_j)}{\det(\tilde{A}_B)}$  where  $\tilde{A}_j$  arises from  $\tilde{A}_B$  by replacing the  $j$ -th column by  $\tilde{b}$ . Thus, we have  $\text{size}(x) \leq n + 2n(\text{size}(\tilde{A}_j) + \text{size}(\tilde{A}_B)) \leq 4n(\text{size}(\tilde{A}_B) + \text{size}(\tilde{b}))$ .

If  $b \in \{e_i, -e_i\}$ , then  $|\det(\tilde{A}_j)|$  is the absolute value of a determinant of a submatrix of  $\tilde{A}_B$ .  $\square$

**Corollary 43** *Let  $\max\{c^t x \mid Ax \leq b\}$  be a feasible bounded linear program with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . Then, there is an optimum (rational) solution  $x$  such that for each non-zero entry  $x_j$  of  $x$ , we have  $|x_j| \geq 2^{-4n(\text{size}(A) + \text{size}(b))}$ .*

**Proof:** According to the proof of the previous proposition there is an optimum solution  $x$  such that for each entry  $x_j$  of  $x$  we have  $\text{size}(x_j) \leq 4n(\text{size}(A) + \text{size}(b))$ . Since every positive number smaller than  $2^{-4n(\text{size}(A) + \text{size}(b))}$  has a size larger than  $4n(\text{size}(A) + \text{size}(b))$ , this proves the claim.  $\square$

## 5.1 Gaussian Elimination

Assume that we want solve an equation system  $Ax = b$ . We can do this by applying the Gaussian Elimination. This algorithm performs three kinds of operations to the matrix  $A$ :

1. Add a multiple of a row to another row.
2. Swap two columns.
3. Swap two rows.

It should be well-known (see e.g. textbooks Hougardy and Vygen [2018] or Korte and Vygen [2018]) that with these steps  $O(mn(\text{rank}(A) + 1))$  elementary arithmetical operations are sufficient to transform  $A$  into an upper (right) triangular matrix. Then it is easy to check if the equation system is feasible, and, in case that it is feasible, to compute a solution. However, in order to show that Gaussian Elimination is a polynomial-time algorithm, we have to show that the numbers that arise during the algorithm aren't too big.

The intermediate matrices that occur during the algorithm are of the type

$$\begin{pmatrix} B & C \\ 0 & D \end{pmatrix}, \tag{36}$$

where  $B$  is an upper triangular matrix. Then, an elementary step of the Gaussian Elimination consist of choosing a non-zero entry of  $D$  (called pivot element; if no such entry exists, we are done) and to swap rows and/or columns such that this element is at position  $(1, 1)$  of  $D$ . Then we add a multiple of the first row of  $D$  to the other rows of  $D$  such that the entry at position  $(1, 1)$  is the only non-zero entry of the first column of  $D$ .

We want to prove that the numbers that occur during the algorithm can be encoded using a polynomial number of bits. We can assume that we don't need any swapping operation because swapping columns or rows doesn't change the numbers in the matrix.

Assume that our current matrix is  $\tilde{A} = \begin{pmatrix} B & C \\ 0 & D \end{pmatrix}$  where  $B$  is a  $k \times k$ -matrix. Then for each entry  $d_{ij}$  of  $D$  we have

$$\det(\tilde{A}_{1, \dots, k, k+i}^{1, \dots, k, k+i}) = d_{ij} \cdot \det(\tilde{A}_{1, \dots, k}^{1, \dots, k}). \quad (37)$$

where  $M_{j_1, \dots, j_t}^{i_1, \dots, i_t}$  denotes the submatrix of a matrix  $M$  induced by the rows  $i_1, \dots, i_t$  and the columns  $j_1, \dots, j_t$ . To see the correctness of (37), apply Laplace's formula to the last row of  $\tilde{A}_{1, \dots, k, k+i}^{1, \dots, k, k+i}$  which contains  $d_{ij}$  as the only non-zero element. Since the determinant does not change if we add the multiple of a row to another row, this leads to

$$d_{ij} = \frac{\det(A_{1, \dots, k, k+i}^{1, \dots, k, k+i})}{\det(A_{1, \dots, k}^{1, \dots, k})}$$

By Proposition 41 and Proposition 39, this implies  $\text{size}(d_{ij}) \leq 4 \text{size}(A)$ . Since all entries of the matrix occur as entries of such a matrix  $D$ , this shows that the sizes of all numbers that are considered during the Gaussian Elimination are bounded by  $4 \text{size}(A)$ .

Note that we have to apply the Euclidean Algorithm to any intermediate result in order to get small representations of the numbers. But this is not a problem because the Euclidean Algorithm is polynomial as well.

Finally, we get the result:

**Proposition 44** *The Gaussian Elimination is an algorithm with polynomial running time.* □

In particular this result shows that the following problems can be solved with a polynomial running time:

- Solving a system of linear equations.
- Computing the determinant of a matrix.
- Computing the rank of a matrix.
- Computing the inverse of a regular matrix.
- Checking if a set of rational vectors is linearly independent.

## 6 Ellipsoid Method

The Ellipsoid Method (proposed by Khachiyan [1979]) was the first polynomial-time algorithm for linear programming. The algorithm solves the problem of finding a feasible solution of a linear program. As we have seen in Section 2.4, this is sufficient to solve as well the optimization problem.

### 6.1 Idealized Ellipsoid Method

**Definition 15** A set  $E \subset \mathbb{R}^n$  is an **ellipsoid** if there are a vector  $s \in \mathbb{R}^n$  and a nonsingular matrix  $M \in \mathbb{R}^{n \times n}$  such that

$$E = \{Mx + s \mid x \in B^n\}$$

where  $B^n = \{x \in \mathbb{R}^n \mid x^t x \leq 1\}$  is the  $n$ -dimensional unit ball.

As a short notation, we write  $E = s + MB^n$ .

**Definition 16** A symmetric matrix  $A$  is called **positive definite** if  $x^t Ax > 0$  for any non-zero vector  $x$ . It is called **positive semidefinite** if  $x^t Ax \geq 0$  for any vector  $x$ .

**Remark:** An  $n \times n$ -matrix  $Q$  is positive definite if and only if there is a non-singular matrix  $M$  such that  $Q = MM^t$ . For example, the **Cholesky decomposition** of  $Q$  achieves this. For a proof of this statement, we refer to textbooks on linear algebra, e.g. Strang [1980].

**Lemma 45** A set  $E \subset \mathbb{R}^n$  is an ellipsoid if and only if there is a (symmetric) positive definite  $n \times n$ -matrix  $Q$  and a vector  $s \in \mathbb{R}^n$  such that  $E = \{x \in \mathbb{R}^n \mid (x-s)^t Q^{-1}(x-s) \leq 1\}$ .

**Proof:** A set  $E \subseteq \mathbb{R}^n$  is an ellipsoid if and only if there is a nonsingular matrix  $M \in \mathbb{R}^{n \times n}$  and a vector  $s \in \mathbb{R}^n$  such that

$$E = \{Mx+s \mid x \in B^n\} = \{y \in \mathbb{R}^n \mid M^{-1}(y-s) \in B^n\} = \{y \in \mathbb{R}^n \mid (y-s)^t (M^{-1})^t M^{-1}(y-s) \leq 1\}.$$

But (using the previous remark) this is equivalent to the statement that there is a positive definite  $n \times n$ -matrix  $Q$  and a vector  $s \in \mathbb{R}^n$  such that  $E = \{x \in \mathbb{R}^n \mid (x-s)^t Q^{-1}(x-s) \leq 1\}$ .  
□

The ELLIPSOID ALGORITHM just finds an element in an polytope or ends with the assertion that the polytope is empty. On the other hand, it can be applied to more general sets  $K \subseteq \mathbb{R}^n$

provided that  $K$  is a compact convex set and that for any  $x \in \mathbb{R}^n \setminus K$  we can find a half-space containing  $K$  such that  $x$  is on the border of the half-space.

Basically, the algorithm works as follows: We always keep track of an ellipsoid containing  $K$ . Then we check if the center  $c$  of the ellipsoid is contained in  $K$ . If this is the case, we are done. Otherwise, we compute the intersection  $X$  of the ellipsoid and a half-space containing  $K$  such that  $c$  is on the border of the half-space. Then, we find a new (smaller) ellipsoid containing  $X$ .

For the 1-dimensional space, the ellipsoid method contains the binary search as a special case. However, for technical reasons, we assume in the following that the dimension of our solution space is at least 2.

We start with a special case that is easier to handle: We assume that our given ellipsoid is the ball  $B^n$  (with radius 1 and center 0). We want to find a small ellipsoid  $E$  covering the intersection of  $B^n$  with the half-space  $\{x \in \mathbb{R}^n \mid x_1 \geq 0\}$  (the gray area in Figure 7).

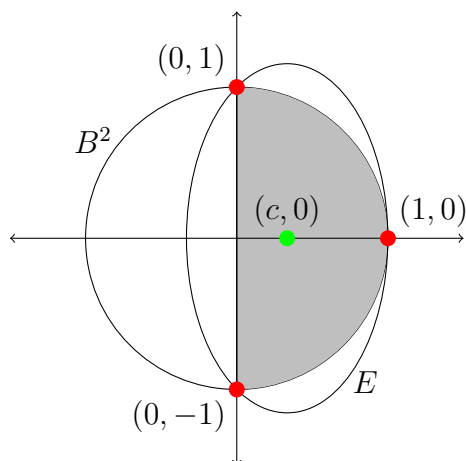


Fig. 7: Intersection of  $B^n$  with  $\{x \in \mathbb{R}^n \mid x_1 \geq 0\}$ .

For symmetry reasons, we choose the center of the new smaller ellipsoid on the vector  $e_1$  at a position  $c \cdot e_1$  (where  $c$  is still to be determined). Our candidates for the ellipsoid are of the form

$$E = \left\{ x \in \mathbb{R}^n \mid \alpha^2(x_1 - c)^2 + \beta^2 \sum_{i=2}^n x_i^2 \leq 1 \right\}$$

where we also have to choose  $\alpha$  and  $\beta$ . The matrix  $Q$  is then a diagonal matrix with entry  $\frac{1}{\alpha^2}$  at position  $(1, 1)$  and  $\frac{1}{\beta^2}$  on all other diagonal positions.

To keep  $E$  small, we want  $e_1$  to lie on the border of  $E$ . This condition leads to  $\alpha^2(1 - c)^2 = 1$  and hence

$$\alpha^2 = \frac{1}{(1 - c)^2}. \quad (38)$$

Moreover, we want all points on the intersection of the border of  $B^n$  and  $\{x \in \mathbb{R}^n \mid x_1 = 0\}$  to



be on the border of  $E$ . This condition leads to  $\alpha^2 c^2 + \beta^2 = 1$  and thus

$$\beta^2 = 1 - \alpha^2 c^2 = 1 - \frac{c^2}{(1-c)^2} = \frac{1-2c}{(1-c)^2}. \quad (39)$$

The volume of an ellipsoid  $E = \{x \in \mathbb{R}^n \mid (x-s)^t Q^{-1} (x-s) \leq 1\}$  is  $\text{vol}(E) = \sqrt{\det(Q)} \times \text{vol}(B^n)$  (a result from measure theory, see e.g. Proposition 6.1.2 in Cohn [1980]).

Therefore, our goal is to choose  $\alpha$ ,  $\beta$  and  $c$  in such a way that  $\sqrt{\det(Q)} = \alpha^{-1} \beta^{-(n-1)}$  is minimized.

Thus, we want to find a  $c$  minimizing  $\frac{(1-c)^{2n}}{(1-2c)^{n-1}}$ .

We have  $\frac{d}{dc} \frac{(1-c)^{2n}}{(1-2c)^{n-1}} = \frac{2(n-1)(1-c)^{2n}}{(1-2c)^n} - \frac{2n(1-c)^{2n-1}}{(1-2c)^{n-1}}$  which is zero if  $\frac{2(n-1)(1-c)}{1-2c} = 2n$ . This leads to  $2(n-1) - 2c(n-1) = 2n - 4cn$  and  $c(2n - (n-1)) = 1$ . Thus, we minimize the volume by setting  $c = \frac{1}{n+1}$ .

Then,  $\alpha^2 = \frac{(n+1)^2}{n^2}$  and  $\beta^2 = \frac{n^2-1}{n^2}$ .

**Lemma 46** (*Half-Ball Lemma*) *We have*

$$B^n \cap \{x \in \mathbb{R}^n \mid x_1 \geq 0\} \subseteq E := \left\{ x \in \mathbb{R}^n \mid \frac{(n+1)^2}{n^2} \left( x_1 - \frac{1}{n+1} \right)^2 + \frac{n^2-1}{n^2} \sum_{i=2}^n x_i^2 \leq 1 \right\}.$$

Moreover,  $\frac{\text{vol}(E)}{\text{vol}(B^n)} \leq e^{-\frac{1}{2(n+1)}}$ .

**Proof:** Consider  $x \in B^n \cap \{x \in \mathbb{R}^n \mid x_1 \geq 0\}$ . We have  $\sum_{i=2}^n x_i^2 \leq 1 - x_1^2$ , and hence it is sufficient to show that  $g(x_1) := \frac{(n+1)^2}{n^2} \left( x_1 - \frac{1}{n+1} \right)^2 + \frac{n^2-1}{n^2} (1 - x_1^2) \leq 1$ . For  $x_1 = 0$ , we have  $g(0) = \frac{(n+1)^2}{n^2} \frac{1}{(n+1)^2} + \frac{n^2-1}{n^2} = 1$ . And for  $x_1 = 1$ :  $g(1) = \frac{(n+1)^2}{n^2} \left( \frac{n}{n+1} \right)^2 = 1$ .

Moreover,  $g$  is a quadratic function and the coefficient of  $x_1^2$  is  $\frac{(n+1)^2}{n^2} - \frac{n^2-1}{n^2} > 0$ . Therefore, we have  $g(x_1) \leq 1$  for  $0 \leq x_1 \leq 1$ .

For the second statement, note that  $\frac{\text{vol}(E)}{\text{vol}(B^n)} = \sqrt{\det(Q)} = \alpha^{-1} \beta^{-(n-1)} = \frac{n}{n+1} \left( \frac{n^2}{n^2-1} \right)^{\frac{n-1}{2}} \leq e^{-\frac{1}{n+1}} e^{\frac{n-1}{2(n^2-1)}} = e^{-\frac{1}{n+1} + \frac{1}{2(n+1)}} = e^{-\frac{1}{2(n+1)}}$ . For the first inequality we made use of the fact that  $1+x \leq e^x$  for any  $x \in \mathbb{R}$ .  $\square$

**Lemma 47** (*Half-Ellipsoid Lemma*) Let  $E = p + \{x \in \mathbb{R}^n \mid x^t Q^{-1} x \leq 1\}$  be an ellipsoid and  $a \in \mathbb{R}^n$  with  $a^t Q a = 1$ . Then,

$$E \cap \{x \in \mathbb{R}^n \mid a^t x \geq a^t p\} \subseteq E' = p + \frac{1}{n+1} Q a + \left\{ x \in \mathbb{R}^n \mid \frac{n^2-1}{n^2} x^t \left( Q^{-1} + \frac{2}{n-1} a a^t \right) x \leq 1 \right\}.$$

Moreover,  $\frac{\text{vol}(E')}{\text{vol}(E)} \leq e^{-\frac{1}{2(n+1)}}$ .

**Proof:** Let  $M$  be a non-singular  $n \times n$ -matrix with  $Q = M M^t$ . We can assume that  $a^t M = e_1^t$  (and thus  $Q a = M M^t a = M (a^t M)^t = M e_1$ ) because otherwise we can multiply  $M$  by a rotation matrix that maps the vector  $a^t M$  to  $e_1$ . Then

$$\begin{aligned} & E \cap \{x \in \mathbb{R}^n \mid a^t x \geq a^t p\} \\ &= (p + M B^n) \cap \{x \in \mathbb{R}^n \mid a^t x \geq a^t p\} \\ &= p + (M B^n \cap \{x \in \mathbb{R}^n \mid a^t(x+p) \geq a^t p\}) \\ &= p + (M B^n \cap \{x \in \mathbb{R}^n \mid a^t x \geq 0\}) \\ &= p + M(B^n \cap M^{-1}\{x \in \mathbb{R}^n \mid a^t x \geq 0\}) \\ &= p + M(B^n \cap \{x \in \mathbb{R}^n \mid a^t M x \geq 0\}) \\ &= p + M(B^n \cap \{x \in \mathbb{R}^n \mid e_1^t x \geq 0\}) \\ &\stackrel{\text{Lem. 46}}{\subseteq} p + \frac{1}{n+1} M e_1 + M \left\{ x \in \mathbb{R}^n \mid \frac{n^2-1}{n^2} x^t \left( I_n + \frac{2}{n-1} e_1 e_1^t \right) x \leq 1 \right\} \\ &= p + \frac{1}{n+1} M e_1 + \left\{ x \in \mathbb{R}^n \mid \frac{n^2-1}{n^2} (M^{-1} x)^t \left( I_n + \frac{2}{n-1} e_1 e_1^t \right) M^{-1} x \leq 1 \right\} \\ &= p + \frac{1}{n+1} Q a + \left\{ x \in \mathbb{R}^n \mid \frac{n^2-1}{n^2} x^t \left( Q^{-1} + \frac{2}{n-1} a a^t \right) x \leq 1 \right\} \end{aligned}$$

We can write the ellipsoid  $E'$  in standard form as  $E' = p + \frac{1}{n+1} Q a + \left\{ x \in \mathbb{R}^n \mid x^t \tilde{Q}^{-1} x \leq 1 \right\}$  with  $\tilde{Q} = \frac{n^2}{n^2-1} \left( Q - \frac{2}{n+1} Q a a^t Q^t \right)$  because

$$\begin{aligned} & \frac{n^2-1}{n^2} \left( Q^{-1} + \frac{2}{n-1} a a^t \right) \frac{n^2}{n^2-1} \left( Q - \frac{2}{n+1} Q a a^t Q^t \right) \\ &= I_n - \frac{2}{n+1} a a^t Q^t + \frac{2}{n-1} a a^t Q - \frac{4}{n^2-1} a \underbrace{a^t Q a}_{=1} a^t Q^t \\ &= I_n. \end{aligned}$$

Therefore,  $\frac{\text{vol}(E')}{\text{vol}(E)} = \sqrt{\frac{\det(\tilde{Q})}{\det(Q)}}$ .

We have  $\frac{\det(\tilde{Q})}{\det(Q)} = \det \left( \frac{n^2}{n^2-1} \left( I_n - \frac{2}{n+1} a a^t Q^t \right) \right) = \left( \frac{n^2}{n^2-1} \right)^n \det \left( I_n - \frac{2}{n+1} a a^t Q^t \right) = \left( \frac{n^2}{n^2-1} \right)^n \left( 1 - \frac{2}{n+1} \right)$ . To see the last equality note that the matrix  $a a^t Q^t$  has eigenvalue 1 for the eigenvector

$a$  (because  $a^t Q^t a = 1$ ) while all other eigenvalues are zero (the rank of  $aa^t Q^t$  is 1). Since the determinant is the product of all eigenvalues, this implies the last equation. Hence,  $\sqrt{\frac{\det(\tilde{Q})}{\det(Q)}} \leq \left(\frac{n^2}{n^2-1}\right)^{\frac{n}{2}} \left(1 - \frac{2}{n+1}\right)^{\frac{1}{2}} = \frac{n}{n+1} \left(\frac{n^2}{n^2-1}\right)^{\frac{n-1}{2}} \leq e^{-\frac{1}{2(n+1)}}$  (see the proof of the Half-Ball Lemma for details of the last steps).  $\square$

**Remark:** The ellipsoid  $E' = p + \frac{1}{n+1} Q a + \left\{x \in \mathbb{R}^n \mid x^t \tilde{Q}^{-1} x \leq 1\right\}$  with  $\tilde{Q} = \frac{n^2}{n^2-1} \left(Q - \frac{2}{n+1} Q a a^t Q\right)$  is called **Löwner-John ellipsoid**. It is in fact the smallest ellipsoid containing  $E \cap \{x \in \mathbb{R}^n \mid a^t x \geq a^t p\}$ .

A **separation oracle** for a convex set  $K \subseteq \mathbb{R}^n$  is a black-box algorithm which, given  $x \in \mathbb{R}^n$ , either returns an  $a \in \mathbb{R}^n$  with  $a^t y > a^t x$  for all  $y \in K$  or asserts  $x \in K$ .

**Observation:** Given  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ , a separation oracle for  $\{x \in \mathbb{R}^n \mid Ax \leq b\}$  can be implemented in  $O(mn)$  arithmetical operations.

---

**Algorithm 2:** Idealized Ellipsoid Algorithm

---

**Input:** A separation oracle for a closed convex set  $K \subseteq \mathbb{R}^n$ , a number  $R > 0$  with  $K \subseteq \{x \in \mathbb{R}^n \mid x^t x \leq R^2\}$ , and a number  $\epsilon > 0$

**Output:** An  $x \in K$  or the message “vol( $K$ ) <  $\epsilon$ ”.

```

1  $p_0 := 0, A_0 := R^2 I_n;$ 
2 for  $k = 0, \dots, N(R, \epsilon) := \lceil 2(n+1)(n \ln(2R) + \ln(\frac{1}{\epsilon})) \rceil$  do
3   if  $p_k \in K$  then
4     return  $p_k;$ 
5   Let  $\bar{a} \in \mathbb{R}^n$  be a vector with  $\bar{a}^t y > \bar{a}^t p_k$  for all  $y \in K;$ 
6    $b_k := \frac{A_k \bar{a}}{\sqrt{\bar{a}^t A_k \bar{a}}};$ 
7    $p_{k+1} := p_k + \frac{1}{n+1} b_k;$ 
8    $A_{k+1} := \frac{n^2}{n^2-1} \left(A_k - \frac{2}{n+1} b_k b_k^t\right);$ 
9 return “vol( $K$ ) <  $\epsilon$ ”;

```

---

**Theorem 48** *Given a convex set  $K \subseteq \mathbb{R}^n$  (specified by a separation oracle),  $\epsilon > 0$ , and a number  $R$  with  $K \subseteq \{x \in \mathbb{R}^n \mid x^t x \leq R^2\}$ , we can find an  $x \in K$  or (correctly) assert “vol( $K$ ) <  $\epsilon$ ”, in  $O(n(n \ln(R) + \ln(\frac{1}{\epsilon})))$  iterations of the IDEALIZED ELLIPSOID METHOD. Each iteration requires one oracle call,  $O(n^2)$  basic arithmetical operations, and the computation of one square root of real numbers.*

**Proof:** As an invariant, we will prove that during the  $k$ -th iteration of the algorithm, the set  $K$  is contained in the set  $p_k + \{x \in \mathbb{R}^n \mid x^t A_k^{-1} x \leq 1\}$ . For  $k = 0$ , this is true because  $R$  is big enough. For the step from  $k$  to  $k + 1$ , we apply the Half-Ellipsoid Lemma (Lemma 47) to  $Q = A_k$  and  $a = \frac{\bar{a}}{\sqrt{\bar{a}^t A_k \bar{a}}}$  (this scaling leads to  $a^t A_k a = \frac{\bar{a}^t A_k \bar{a}}{\bar{a}^t A_k \bar{a}} = 1$ ).

We have  $\text{vol}(\{x \in \mathbb{R}^n \mid x^t x \leq R^2\}) \leq \text{vol}([-R, R]^n) = 2^n R^n$ , and in each iteration, the volume of  $E_k = \{x \in \mathbb{R}^n \mid x^t A_k^{-1} x \leq 1\}$  is reduced at least by the factor  $e^{-\frac{1}{2(n+1)}}$ , so we get  $\text{vol}(E_k) \leq e^{-\frac{k}{2(n+1)}} 2^n R^n$ .

Thus, we have to find a smallest  $k$  such that  $e^{-\frac{k}{2(n+1)}} 2^n R^n \leq \epsilon$  which is equivalent to  $\frac{k}{2(n+1)} \geq \ln\left(\frac{2^n R^n}{\epsilon}\right)$  and  $k \geq 2(n+1)(n \ln(2R) + \ln(\frac{1}{\epsilon}))$ . This shows that  $O(n(n \ln(R) + \ln(\frac{1}{\epsilon})))$  iterations are sufficient.  $\square$

## 6.2 Error Analysis

We cannot compute square roots exactly, so during the algorithm, we have to work with rounded intermediate solutions. Let  $\tilde{p}_k$  and  $\tilde{A}_k$  be the exact values and  $p_k$  and  $A_k$  be the rounded values (and the same for the corresponding ellipsoids  $\tilde{E}_k$  and  $E_k$ ). Note that  $\tilde{p}_k$  and  $\tilde{A}_k$  are based on the rounded values  $p_{k-1}$  and  $A_{k-1}$ .

Let  $\delta$  be an upper bound on the maximum absolute rounding error for the entries in  $\tilde{p}_k$  and  $\tilde{A}_k$ , so  $\|p_k - \tilde{p}_k\|_\infty \leq \delta$  and  $\|A_k - \tilde{A}_k\|_\infty \leq \delta$ . So  $\delta$  (that will be defined later) describes the precision of the rounding. When we round the entries in  $\tilde{A}_k$ , we do it in such a way that the matrix remains symmetric. Let  $\Gamma_k = A_k - \tilde{A}_k$  and  $\Delta_k = p_k - \tilde{p}_k$ .

In the following, we write by  $\|\cdot\|$  the Euclidean norm for vectors and the induced operator norm for the matrices. When considering matrices, we often make use of the fact that the Frobenius norm is an upper bound for the operator norm induced by the Euclidean norm.

For any  $x \in K$  we can assume that  $(x - \tilde{p}_k)^t \tilde{A}_k^{-1} (x - \tilde{p}_k) \leq 1$  and we want to prove the same for  $p_k$  and  $A_k$ . To this end, we have to increase the ellipsoid slightly by scaling  $\tilde{A}_k$ .

We have  $(x - p_k)^t A_k^{-1} (x - p_k) = (x - p_k)^t \tilde{A}_k^{-1} (x - p_k) + (x - p_k)^t (A_k^{-1} - \tilde{A}_k^{-1}) (x - p_k)$ . We analyze the two summands separately:

$$\begin{aligned} (x - p_k)^t \tilde{A}_k^{-1} (x - p_k) &= (x - \tilde{p}_k)^t \tilde{A}_k^{-1} (x - \tilde{p}_k) + |2\Delta_k^t \tilde{A}_k^{-1} (x - \tilde{p}_k)| + \Delta_k^t \tilde{A}_k^{-1} \Delta_k \\ &\leq 1 + 2\|\Delta_k\| \cdot \|\tilde{A}_k^{-1}\| (R + \|\tilde{p}_k\|) + \|\Delta_k\|^2 \cdot \|\tilde{A}_k^{-1}\| \\ &\leq 1 + 2\sqrt{n}\delta \|\tilde{A}_k^{-1}\| (R + \|\tilde{p}_k\|) + n\delta^2 \|\tilde{A}_k^{-1}\|. \end{aligned} \quad (40)$$

And:

$$\begin{aligned} (x - p_k)^t (A_k^{-1} - \tilde{A}_k^{-1}) (x - p_k) &\leq \|x - p_k\|^2 \cdot \|A_k^{-1} - \tilde{A}_k^{-1}\| \\ &\leq (R + \|p_k\|)^2 \|A_k^{-1} (A_k - \tilde{A}_k) \tilde{A}_k^{-1}\| \\ &\leq (R + \|p_k\|)^2 \|A_k^{-1}\| \cdot \|\tilde{A}_k^{-1}\| \cdot \|\Gamma_k\| \\ &\leq (R + \|p_k\|)^2 \|A_k^{-1}\| \cdot \|\tilde{A}_k^{-1}\| \cdot n\delta \end{aligned} \quad (41)$$

We adjust  $\tilde{A}_k$  by multiplying it by  $\mu = 1 + \frac{1}{2n(n+1)}$ , so we replace  $\tilde{A}_k$  by  $\mu\tilde{A}_k$  (which we call  $\tilde{A}_k$  again). Then

$$(x - \tilde{p}_k)^t \tilde{A}_k^{-1} (x - \tilde{p}_k) = \frac{1}{1 + \frac{1}{2n(n+1)}} = \frac{2n(n+1)}{2n^2 + 2n + 1} < 1 - \frac{1}{4n^2}. \quad (42)$$

and  $(\widetilde{E}_{k+1})$  also refers to the scaled version of  $\widetilde{A}_k$ :

$$\frac{\text{vol}(\widetilde{E}_{k+1})}{\text{vol}(E_k)} \leq e^{-\frac{1}{2(n+1)}} \left(1 + \frac{1}{2n(n+1)}\right)^{\frac{n}{2}} \leq e^{-\frac{1}{2(n+1)}} e^{\frac{1}{4(n+1)}} = e^{-\frac{1}{4(n+1)}}. \quad (43)$$

Thus,

$$\frac{\text{vol}(E_{k+1})}{\text{vol}(E_k)} = \frac{\text{vol}(\widetilde{E}_{k+1}) \text{vol}(E_{k+1})}{\text{vol}(E_k) \text{vol}(\widetilde{E}_{k+1})} \leq e^{-\frac{1}{4(n+1)}} \sqrt{\det(A_{k+1} \widetilde{A}_{k+1}^{-1})} \quad (44)$$

We have

$$\begin{aligned} \det(A_{k+1} \widetilde{A}_{k+1}^{-1}) &= \det\left(I_n + (A_{k+1} - \widetilde{A}_{k+1}) \widetilde{A}_{k+1}^{-1}\right) \\ &\stackrel{(*)}{\leq} \|I_n + (A_{k+1} - \widetilde{A}_{k+1}) \widetilde{A}_{k+1}^{-1}\|^n \\ &\leq (1 + \|\Gamma_{k+1}\| \cdot \|\widetilde{A}_{k+1}^{-1}\|)^n \\ &\leq (1 + n\delta \|\widetilde{A}_{k+1}^{-1}\|)^n \\ &\leq e^{n^2\delta \|\widetilde{A}_{k+1}^{-1}\|}, \end{aligned}$$

where inequality  $(*)$  follows from Hadamard's inequality ( $|\det(A)| \leq \prod_{i=1}^n \|a_i\|$  for an  $n \times n$ -matrix with columns  $a_1, \dots, a_n$ , see exercises).

This implies

$$\frac{\text{vol}(E_{k+1})}{\text{vol}(E_k)} \leq e^{-\frac{1}{4(n+1)}} \cdot e^{\frac{1}{2}n^2\delta \|\widetilde{A}_{k+1}^{-1}\|}.$$

Hence, if we had  $\frac{1}{2}\delta \|\widetilde{A}_{k+1}^{-1}\| < \frac{1}{8(n+1)^3}$ , then we had  $\frac{\text{vol}(E_{k+1})}{\text{vol}(E_k)} < e^{-\frac{1}{8(n+1)}}$ .

Therefore, and by equations (40) and (41) our goal is to choose  $\delta$  such that we get the following inequalities:

- $2\sqrt{n}\delta \|\widetilde{A}_k^{-1}\| (R + \|\tilde{p}_k\|) + n\delta^2 \|\widetilde{A}_k^{-1}\| + (R + \|p_k\|)^2 \|A_k^{-1}\| \cdot \|\widetilde{A}_k^{-1}\| n\delta \leq \frac{1}{4n^2}$
- $\delta \|\widetilde{A}_{k+1}^{-1}\| \leq \frac{1}{4(n+1)^3}$

For the analysis, we assume that  $R \geq 1$ .

**Proposition 49** *Assume that  $\delta$  is chosen such that  $\delta \leq \frac{1}{12n4^k}$  in iteration  $k$  of the ELLIPSOID METHOD. Then, we have:*

- (a)  $A_k$  is positive definite.
- (b)  $\|p_k\| \leq R2^k$ ,  $\|\tilde{p}_k\| \leq R2^k$ .
- (c)  $\|A_k\| \leq R^2 2^k$ ,  $\|\widetilde{A}_k\| \leq R^2 2^k$ .
- (d)  $\|A_k^{-1}\| \leq R^{-2} 4^k$ ,  $\|\widetilde{A}_k^{-1}\| \leq R^{-2} 4^k$ .

**Proof:** We have

$$\widetilde{A_{k+1}}^{-1} = \frac{n^2 - 1}{n^2} \frac{1}{\mu} \left( A_k^{-1} + \frac{2}{n-1} \frac{\bar{a}\bar{a}^t}{\bar{a}^t A_k \bar{a}} \right).$$

Thus, as a sum of a positive definite matrix and a positive semidefinite matrix  $\widetilde{A_{k+1}}^{-1}$  is positive definite. Therefore  $\widetilde{A_{k+1}} = \frac{n^2}{n^2-1} \mu (A_k - \frac{2}{n-1} b_k b_k^t)$  is positive definite.

We will show by induction that  $A_k$  is positive definite and  $\|A_k^{-1}\| \leq R^{-2} 4^k$ .

$$\left\| \frac{\bar{a}\bar{a}^t}{\bar{a}^t A_k \bar{a}} \right\| = \frac{\bar{a}^t \bar{a}}{\bar{a}^t A_k \bar{a}} \leq (\min\{x^t A_k x \mid \|x\| = 1\})^{-1} \leq \|A_k^{-1}\|.$$

Thus,

$$\|\widetilde{A_{k+1}}^{-1}\| \leq \frac{n^2 - 1}{n^2} \frac{1}{\mu} \left( \|A_k^{-1}\| + \frac{2}{n-1} \left\| \frac{\bar{a}\bar{a}^t}{\bar{a}^t A_k \bar{a}} \right\| \right) \leq 3 \|A_k^{-1}\|$$

Let  $\lambda$  be a smallest eigenvalue of  $A_{k+1}$  and  $v$  a vector with  $\|v\| = 1$  such that  $\lambda = v^t A_{k+1} v$ . Then:

$$\begin{aligned} v^t A_{k+1} v &\geq v^t \widetilde{A_{k+1}} v - n\delta \\ &\geq \min\{u^t \widetilde{A_{k+1}} u \mid u \in \mathbb{R}^n, \|u\| = 1\} - n\delta \\ &\geq \frac{1}{\|\widetilde{A_{k+1}}^{-1}\|} - n\delta \\ &\geq \frac{1}{3\|A_k^{-1}\|} - n\delta \\ &\geq \frac{1}{3} \frac{1}{R^{-2} 4^k} - n\delta \\ &\geq \frac{1}{R^{-2} 4^{k+1}}, \end{aligned}$$

provided that:

$$n\delta \leq \left( \frac{1}{3} - \frac{1}{4} \right) \frac{R^2}{4^k}. \quad (45)$$

This shows that  $A_{k+1}$  is positive definite and by  $\|A_0^{-1}\| = R^{-2}$  and  $\frac{1}{\|A_{k+1}^{-1}\|} = v^t A_{k+1} v$  this proves  $\|A_{k+1}^{-1}\| \leq R^{-2} 4^{k+1}$

By  $\|\widetilde{A_{k+1}}^{-1}\| \leq 3\|A_k^{-1}\|$  we get as well  $\|\widetilde{A_{k+1}}^{-1}\| \leq R^{-2} 4^{k+1}$ . This proves (d).

We have  $\|\widetilde{A_{k+1}}\| \leq \frac{n^2}{n^2-1} \mu \|A_k\|$  because  $\|A\| \leq \|A + B\|$  for positive semidefinite matrices  $A$  and  $B$  (see the exercises). Together with  $\|A_0\| = R^2$ , this leads by induction to

$$\|A_{k+1}\| \leq \|\widetilde{A_{k+1}}\| + \|\Gamma_{k+1}\| \leq \underbrace{\frac{n^2}{n^2-1} \mu}_{\leq \frac{3}{2}} \|A_k\| + n\delta \leq R^2 2^{k+1}$$

We also get  $\|\widetilde{A_{k+1}}\| \leq \frac{n^2}{n^2-1} \mu \|A_k\| \leq R^2 2^{k+1}$ , so we have proved (c).

We can write  $A_k = MM^t$  with a regular matrix  $M$ . Then,

$$\|b_k\| = \frac{\|A_k \bar{a}\|}{\sqrt{\bar{a}^t A_k \bar{a}}} = \sqrt{\frac{\bar{a}^t A_k A_k \bar{a}}{\bar{a}^t A_k \bar{a}}} = \sqrt{\frac{(M^t \bar{a})^t A_k (M^t \bar{a})}{(M^t \bar{a})^t (M^t \bar{a})}} \leq \sqrt{\|A_k\|} \leq R2^{\frac{k}{2}}, \quad (46)$$

where the first inequality follows from the fact that  $\|A_k\| = \max\{x^t A_k x \mid \|x\| = 1\}$  because  $A_k$  is positive semidefinite (see exercises).

Therefore, we get by induction (using the fact that  $p_0 = 0$ )

$$\|p_{k+1}\| \leq \|p_k\| + \frac{1}{n+1} \|b_k\| + \sqrt{n}\delta \leq \|p_k\| + R2^{\frac{k}{2}} + \sqrt{n}\delta \leq R2^k + R2^{\frac{k}{2}} + \frac{1}{3\sqrt{n}4^k} \leq R2^{k+1}.$$

This also gives us:  $\|\widetilde{p_{k+1}}\| \leq \|p_k\| + \frac{1}{n+1} \|b_k\| \leq R2^{k+1}$ . This shows statement (b).  $\square$

---

**Algorithm 3:** Ellipsoid Algorithm

---

**Input:** A separation oracle for a closed convex set  $K \subseteq \mathbb{R}^n$ , a number  $R > 0$  with  $K \subseteq \{x \in \mathbb{R}^n \mid x^t x \leq R^2\}$ , and a number  $\epsilon > 0$

**Output:** An  $x \in K$  or the message “vol( $K$ ) <  $\epsilon$ ”.

- 1  $p_0 := 0, A_0 := R^2 I_n$ ;
  - 2 **for**  $k = 0, \dots, N(R, \epsilon) := \lceil 8(n+1)(n \ln(2R) + \ln(\frac{1}{\epsilon})) \rceil$  **do**
  - 3     **if**  $p_k \in K$  **then**
  - 4         **return**  $p_k$ ;
  - 5     Let  $\bar{a} \in \mathbb{R}^n$  be a vector with  $\bar{a}^t y > \bar{a}^t p_k$  for all  $y \in K$ ;
  - 6      $b_k := \frac{A_k \bar{a}}{\sqrt{\bar{a}^t A_k \bar{a}}}$ ;
  - 7      $p_{k+1}$  an approximation of  $\widetilde{p_{k+1}} := p_k + \frac{1}{n+1} b_k$  with a maximum error of  $\delta$ ;
  - 8      $A_{k+1}$  a symmetric approximation of  $\widetilde{A_{k+1}} := \left(1 + \frac{1}{2n(n+1)}\right) \frac{n^2}{n^2-1} (A_k - \frac{2}{n+1} b_k b_k^t)$  with a maximum error of  $\delta$ ;
  - 9 **return** “vol( $K$ ) <  $\epsilon$ ”;
- 

**Lemma 50** Let  $\delta$  be positive with  $\delta < (2^{6(N(R,\epsilon)+1)} 16n^3)^{-1}$  where  $N(R, \epsilon) := \lceil 8(n+1)(n \ln(2R) + \ln(\frac{1}{\epsilon})) \rceil$ . Then, in iteration  $k$  of the ELLIPSOID ALGORITHM, we have  $K \subseteq p_k + E_k$  and  $\text{vol}(E_k) < e^{-\frac{k}{8(n+1)}} 2^n R^n$ .

**Proof:** By the choice of  $\delta$ , we have  $n\delta \leq (\frac{1}{3} - \frac{1}{4}) \frac{R^2}{4^k}$ .

Moreover,

$$\bullet 2\sqrt{n}\delta \underbrace{\|\widetilde{A_k}^{-1}\|}_{\leq R^{-2 \cdot 4^k}} (R + \underbrace{\|\widetilde{p_k}\|}_{\leq R2^k}) + n\delta^2 \underbrace{\|\widetilde{A_k}^{-1}\|}_{\leq R^{-2 \cdot 4^k}} + (R + \underbrace{\|p_k\|}_{\leq R2^k})^2 \underbrace{\|A_k^{-1}\|}_{\leq R^{-2 \cdot 4^k}} \cdot \underbrace{\|\widetilde{A_k}^{-1}\|}_{\leq R^{-2 \cdot 4^k}} n\delta \leq \delta n 2^{6k} \leq \frac{1}{4n^2}$$

$$\bullet \delta \underbrace{\| \widetilde{A_{k+1}}^{-1} \|}_{\leq R^{-24^k}} \leq \frac{1}{4(n+1)^3}$$

Hence, by the above analysis,  $E_k$  (with rounded numbers) always contains the set  $K$ , and the volume of  $E_k$  is reduced at least by a factor of  $e^{-\frac{1}{8(n+1)}}$  in each iteration, so after  $O(n(n \ln R + \ln(\frac{1}{\epsilon})))$  iterations, the algorithm terminates with a correct output.  $\square$

**Theorem 51** *For a compact convex set  $K \subseteq \{x \in \mathbb{R}^n \mid x^t x \leq R^2\}$ , given by a separation oracle, the ELLIPSOID ALGORITHM either finds a vector  $x \in K$  or asserts  $\text{vol}(K) \leq \epsilon$ . It needs  $O(n(n \ln R + \ln(\frac{1}{\epsilon})))$  iterations, and in each iteration it performs one oracle call, the approximative computation of one square root and  $O(n^2)$  arithmetical operations on  $O(n(n \ln R + \ln(\frac{1}{\epsilon})))$  bits.  $\square$*

There number of calls of the separation oracle can be reduced to  $O(n \ln(\frac{nR}{\epsilon}))$  (see Lee, Sidford, and Wong [2015] for an algorithm that only needs  $O(n \ln(\frac{nR}{\epsilon}))$  oracle calls and  $O(n^3 \ln^{O(1)}(\frac{nR}{\epsilon}))$  additional time).

### 6.3 Ellipsoid Method for Linear Programs

We first want to use the ELLIPSOID ALGORITHM just to check if a given polyhedron  $P$  is empty. This can be done directly, provided that  $P$  is in fact a polytope and if we have the assertion that if  $P$  is non-empty, its volume cannot be arbitrarily small. The following proposition implies that we can assume these properties:

**Proposition 52** *Let  $A \in \mathbb{Q}^{m \times n}$ ,  $b \in \mathbb{Q}^m$  and  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . For  $R = 1 + 2^{4n(\text{size}(A) + \text{size}(b))}$  and  $\epsilon = (2n2^{4n(\text{size}(A) + \text{size}(b))})^{-1}$  let  $P_{R,\epsilon} = \{x \in [-R, R]^n \mid Ax \leq b + \epsilon \mathbb{1}\}$ . Then:*

- (a)  $P = \emptyset \Leftrightarrow P_{R,\epsilon} = \emptyset$ .
- (b) If  $P \neq \emptyset$ , then  $\text{vol}(P_{R,\epsilon}) \geq \left(\frac{2\epsilon}{n2^{\text{size}(A)}}\right)^n$ .

**Proof:**

- (a) “ $P = \emptyset \Rightarrow P_{R,0} = \emptyset$ ” is trivial, and by Proposition 42, we have “ $P_{R,0} = \emptyset \Rightarrow P = \emptyset$ ”. “ $P_{R,\epsilon} = \emptyset \Rightarrow P_{R,0} = \emptyset$ ” is also trivial, so it remains to show: “ $P = \emptyset \Rightarrow P_{R,\epsilon} = \emptyset$ ”. Assume that  $P = \emptyset$ . By Farkas’ Lemma (Theorem 5) this implies that there is a vector  $y \geq 0$  with



$y^t A = 0$  and  $y^t b = -1$ . Then, by Proposition 42

$$\begin{aligned} \min \mathbb{1}^t y \\ A^t y &= 0 \\ b^t y &= -1 \\ y &\geq 0 \end{aligned}$$

has an optimum solution  $y$  such that the absolute value of any entry of  $y$  is at most  $2^{4n \text{size}(A) + \text{size}(b)}$ . Thus,  $y^t(b + \epsilon \mathbb{1}) < -1 + (n + 1)2^{4n(\text{size}(A) + \text{size}(b))} \epsilon < 0$ . Again by Farkas' Lemma, this implies that  $Ax \leq b + \epsilon \mathbb{1}$  does not have a feasible solution. In particular, there is no feasible solution in  $[-R, R]^n$ , so  $P_{R, \epsilon} = \emptyset$ .

- (b) If  $P \neq \emptyset$ , then  $P_{R-1, 0} \neq \emptyset$  (with the same proof as in (a) for  $R$ ). But for any  $z \in P_{R-1, 0}$ , we have  $\{x \in \mathbb{R}^n \mid \|x - z\|_\infty < \frac{\epsilon}{n2^{\text{size}(A)}}\} \subseteq P_{R, \epsilon}$ . Hence  $\text{vol}(P_{R, \epsilon}) \geq \text{vol}\{x \in \mathbb{R}^n \mid \|x - z\|_\infty < \frac{\epsilon}{n2^{\text{size}(A)}}\} = \left(\frac{2\epsilon}{n2^{\text{size}(A)}}\right)^n$ .  $\square$

**Theorem 53** *Given a polyhedron  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$  we can decide in polynomial running time if  $P$  is empty.*

**Proof:** We can apply the ELLIPSOID ALGORITHM to  $K = P_{R, \epsilon}$  with  $R = \lceil \sqrt{n}(1 + 2^{4n(\text{size}(A) + \text{size}(b))}) \rceil$  and  $\epsilon' = \left(\frac{2\epsilon}{n2^{\text{size}(A)}}\right)^n$  (for  $\epsilon = (2n2^{4n(\text{size}(A) + \text{size}(b))})^{-1}$ ) as a lower bound for the volume. We need  $N(R, \epsilon') = O(n(n \ln(R) + \ln(\frac{1}{\epsilon'})))$  iterations, which is polynomial in the input size.

Moreover, it is sufficient to set the bound on the absolute rounding error to any value  $\delta < (2^{6(N(R, \epsilon') + 1)} 16n^3)^{-1}$ , so also the number of bits that we have to compute during the algorithm is polynomial.  $\square$

**Theorem 54** *There is a polynomial-time algorithm that computes an optimum solution for a given linear program  $\max\{c^t x \mid Ax \leq b\}$  with  $A \in \mathbb{Q}^{m \times n}$ ,  $c \in \mathbb{Q}^n$  and  $b \in \mathbb{Q}^m$  if one exists.*

**Proof:** By Theorem 53, we can check in polynomial time if a given linear program has a feasible solution. We will show that this is sufficient for computing a feasible solution if one exists. Assume that we are given  $m$  inequalities  $a_i^t x \leq b_i$  with  $a_i \in \mathbb{Q}^n$  and  $b_i \in \mathbb{Q}$  ( $i \in \{1, \dots, m\}$ ). First check if the system is feasible. If it infeasible, we are done. Otherwise, perform for  $i = 1, \dots, m$  the following steps: Check if the system remains feasible if we replace  $a_i^t x \leq b_i$  by  $a_i^t x = b_i$ . If this is the case, replace  $a_i^t x \leq b_i$  by  $a_i^t x = b_i$ . Otherwise, the inequality is redundant, and we can skip it. We end up with a feasible system of equations with the property that any solution of this system of equations is also a solution of the given system of inequalities. However, the system of equations can be solved in polynomial time by using Gaussian Elimination (see

Section 5.1). Hence, for any linear program, we can compute in polynomial-time a feasible solution if one exists.

In Section 2.4 we have seen that the task of computing an optimum solution for a bounded feasible linear program can be reduced to the computation of a feasible solution of a modified linear program (see the LP (24)). Thus, we can also compute an optimum solution.  $\square$

**Remark:** By Proposition 22, the method described in the previous proof computes a solution in a minimal face of the solution polyhedron  $P$ . In particular, if  $P$  is pointed, we compute a vertex of  $P$ .

## 6.4 Separation and Optimization

An advantage of the ELLIPSOID ALGORITHM is that it does not necessarily need a complete description of a solution space  $K \subseteq \mathbb{R}^n$  but only needs a separation oracle that provides a linear inequality satisfied by all elements of  $K$  but not by a given vector  $x \in \mathbb{R}^n \setminus K$ . This allows us to use the method e.g. for linear program with an exponential number of constraints.

**Example:** Consider the MAXIMUM-MATCHING PROBLEM. A **matching** in an undirected graph is a set  $M \subseteq E(G)$  such that  $|\delta_G(v) \cap M| \leq 1$  for all  $v \in V(G)$ . In the MAXIMUM-MATCHING PROBLEM we are given an undirected graph  $G$  and ask for a matching with maximum cardinality. It can be formulated as the following integer linear program:

$$\begin{aligned} \max \quad & \sum_{e \in E(G)} x_e \\ & \sum_{e \in \delta_G(v)} x_e \leq 1 \quad v \in V(G) \\ & x_e \in \{0, 1\} \quad e \in E(G) \end{aligned}$$

In the LP-relaxation, we simply replace the constraint “ $x_e \in \{0, 1\}$ ” by “ $x_e \geq 0$ ”. However, this allows us e.g. in the graph  $K_3$  (i.e. the complete graph on three vertices) to set all values  $x_e$  to  $\frac{1}{2}$ . To avoid such solutions, we may add the following constraints:

$$\sum_{e \in E(G[U])} x_e \leq \frac{|U|-1}{2} \quad U \subseteq V(G), |U| \text{ odd}$$

It turns out that the feasible solutions of the LP

$$\begin{aligned} \max \quad & \sum_{e \in E(G)} x_e \\ & \sum_{e \in \delta_G(v)} x_e \leq 1 \quad v \in V(G) \\ & \sum_{e \in E(G[U])} x_e \leq \frac{|U|-1}{2} \quad U \subseteq V(G), |U| \text{ odd} \\ & x_e \geq 0 \quad e \in E(G) \end{aligned}$$

are indeed the convex combinations of the solutions of the ILP formulation. In other words, the vertices of the solution polyhedron of the LP are the integer solutions. We won't prove this statement here, see Edmonds [1965] for a proof. Hence, solving the linear program would be sufficient to solve the matching problem. The number of constraints is exponential in the size of the graph, but the good news is that there is a separation oracle with polynomial running

time for this linear program (see Padberg and Rao [1982]). We will see how such a separation oracle can be used for solving the optimization problem.

In the remainder of this chapter, we always consider closed convex sets  $K$  for which numbers  $r$  and  $R$  with  $0 < r < \frac{R}{2}$  exist such that  $rB^n \subseteq K \subseteq RB^n$ . We call sets for which such numbers  $r$  and  $R$  exist,  **$r$ - $R$ -sandwiched sets**.

We will consider relaxed versions both of linear optimization problems and of separation problems. In the **weak optimization problem** we are given a set  $K \subseteq \mathbb{R}^n$ , a number  $\epsilon > 0$  and a vector  $c \in \mathbb{Q}^n$ . The task is to find an  $x \in K$  with  $c^t x \geq \max\{c^t z \mid z \in K\} - \epsilon$ .

In order to apply the Ellipsoid Algorithm directly to an optimization problem, we need the property that the set of almost optimum solutions cannot have an arbitrarily small volume. The following lemma guarantees this for  $r$ - $R$ -sandwiched sets:

**Lemma 55** *Let  $K \subseteq \mathbb{R}^n$  be an  $r$ - $R$ -sandwiched convex set,  $c \in \mathbb{R}^n$ ,  $\delta = \sup\{c^t x \mid x \in K\}$ , and  $0 < \epsilon < \delta$ . Moreover, let  $U = \{x \in K \mid c^t x \geq \delta - \epsilon\}$ . Then,*

$$\text{vol}(U) \geq \left(\frac{\epsilon}{2\|c\|R}\right)^{n-1} r^{n-1} \frac{1}{n^n} \frac{\epsilon}{2\|c\|} \frac{1}{n}.$$

**Proof:** Let  $z \in K$  with  $c^t z \geq \delta - \frac{\epsilon}{2}$ . The set  $A = \{x \in \mathbb{R}^n \mid c^t x = 0, x^t x \leq r^2\}$  is an  $(n-1)$ -dimensional ball of radius  $r$  and is contained in  $K$ . Its  $(n-1)$ -dimensional volume is  $r^{n-1} \text{vol}(B_{n-1})$ . And by convexity of  $K$ , we have  $\text{conv}(A \cup \{z\}) \subseteq K$ . Let  $A' = \text{conv}(A \cup \{z\}) \cap \{x \in \mathbb{R}^n \mid c^t x = c^t z - \frac{\epsilon}{2}\}$ . Then the  $(n-1)$ -dimensional volume of  $A'$  is

$$\left(\frac{\epsilon}{2c^t z}\right)^{n-1} r^{n-1} \text{vol}(B_{n-1})$$

Moreover,  $\text{conv}(A' \cup \{z\}) \subset U$  and

$$\begin{aligned} \text{vol}(\text{conv}(A' \cup \{z\})) &\geq \left(\frac{\epsilon}{2c^t z}\right)^{n-1} r^{n-1} \text{vol}(B_{n-1}) \frac{\epsilon}{2\|c\|} \frac{1}{n} \\ &\geq \left(\frac{\epsilon}{2\|c\|R}\right)^{n-1} r^{n-1} \frac{1}{n^n} \frac{\epsilon}{2\|c\|} \frac{1}{n}. \end{aligned}$$

Here we use the fact that  $\text{conv}(A' \cup \{z\})$  is an  $n$ -dimensional pyramid with height at least  $\frac{\epsilon}{2\|c\|}$  and a base of  $((n-1)$ -dimensional) volume  $\left(\frac{\epsilon}{2c^t z}\right)^{n-1} r^{n-1} \text{vol}(B_{n-1})$ .  $\square$

This result allows us to find a polynomial-time algorithm for the weak optimization problem provided that we can solve the corresponding separation problem efficiently:

**Proposition 56** *Given a polynomial-time separation oracle for an  $r$ - $R$ -sandwiched convex set  $K \subseteq \mathbb{R}^n$  with running time polynomial in  $\text{size}(R)$ ,  $\text{size}(r)$  and  $\text{size}(x)$  (where  $x$  is the input vector for the oracle), a number  $\epsilon > 0$  and a vector  $c$ , there is a polynomial-time algorithm (w.r.t.  $\text{size}(R)$ ,  $\text{size}(r)$ ,  $\text{size}(c)$  and  $\text{size}(\epsilon)$ ) that computes a vector  $v \in K$  with  $c^t v \geq \sup\{c^t x \mid x \in K\} - \epsilon$ .*

**Proof:** Apply the ELLIPSOID ALGORITHM to find an almost optimum vector in  $K$ . Use the previous lemma that shows that the set of almost optimum vectors in  $K$  cannot be arbitrarily small.  $\square$

A **weak separation oracle** for a convex set  $K \subseteq \mathbb{R}^n$  is an algorithm which, given  $x \in \mathbb{R}^n$  and  $\eta$  with  $0 < \eta < \frac{1}{2}$ , either asserts  $x \in K$  or finds  $v \in \mathbb{R}^n$  with  $v^t z \leq 1$  for all  $z \in K$  and  $v^t x \geq 1 - \eta$ .

**Remark:** For the previous proposition, it would be enough to have a weak separation oracle for  $K$ .

**Notation:** For  $K \subseteq \mathbb{R}^n$ , we define  $K^* := \{y \in \mathbb{R}^n \mid y^t x \leq 1 \text{ for all } x \in K\}$ .

**Theorem 57** *If there is an algorithm with running time polynomial in  $\text{size}(r)$  and  $\text{size}(R)$  maximizing linear objective functions over a closed convex  $r$ - $R$ -sandwiched set  $K \subseteq \mathbb{R}^n$ , then there is a weak separation oracle for  $K$  with running time polynomial in  $\text{size}(r)$ ,  $\text{size}(R)$  and  $\text{size}(\eta)$ .*

**Proof:** Claim:  $K^{**} = K$

Proof of the claim: For  $x \in K$ , we have  $y^t x \leq 1$  for all  $y \in K^*$  which implies  $x \in K^{**}$ . Therefore, we have  $K \subseteq K^{**}$ .

Now let  $z \in \mathbb{R}^n \setminus K$ . And let  $w \in K$  be a vector such that  $\|z - w\|_2$  is smallest possible over vectors in  $K$  ( $w$  exists because  $K$  is convex and closed). Let  $u = z - w$ . Then, for all  $x \in K$ , we have  $u^t x \leq u^t w < u^t z$ . Moreover, since  $0 \in K$ , we have  $u^t w \geq 0$ . By scaling  $u$ , we can assume that  $u^t z > 1$  while  $u^t x \leq 1$  for all  $x \in K$ . But then  $u \in K^*$  and  $u^t z > 1$  which implies  $z \notin K^{**}$ . Thus  $K^{**} \subseteq K$ . This prove the claim.

Now, let  $x \in \mathbb{R}^n$  be an instance for the weak separation oracle. If  $x = 0$ , we can assert  $x \in K$ , and if  $\|x\| > R$  we can choose  $v = \frac{x}{\|x\|^2}$ . Therefore, we can assume that  $0 < \|x\| \leq R$ .

We can solve the (strong) separation problem for  $K^*$  (see the exercises). Since  $K^*$  is a closed convex  $\frac{1}{R}$ - $\frac{1}{r}$ -sandwiched set, we can apply the previous observation to it, and thus, we can solve the weak optimization problem for  $K^*$  with  $c = \frac{x}{\|x\|^2}$  and  $\epsilon = \frac{\eta}{R}$  in polynomial time. Thus, we get a vector  $v_0 \in K^*$  with  $\frac{x^t}{\|x\|} v_0 \geq \max\{\frac{x^t}{\|x\|} v \mid v \in K^*\} - \frac{\eta}{R}$ . If  $\frac{x^t}{\|x\|} v_0 \geq \frac{1}{\|x\|} - \frac{\eta}{R}$ , then  $v_0^t x \geq 1 - \eta \frac{\|x\|}{R} \geq 1 - \eta$ , and  $v_0^t z \leq 1$  for all  $z \in K$  (since  $v_0 \in K^*$ ). Otherwise

$\max\{\frac{x^t}{\|x\|}v \mid v \in K^*\} \leq \frac{1}{\|x\|}$ , so  $\max\{x^t v \mid v \in K^*\} \leq 1$ , which implies  $x \in K^{**}$ . Together with the above claim, this implies  $x \in K$ . Therefore, we have a weak separation oracle for  $K$  in polynomial running time.  $\square$

It turns out that for rational  $r$ - $R$ -sandwiched polyhedra  $P$  an exact polynomial-time separation algorithm also provides an exact polynomial-time optimization algorithm, provided that appropriate bounds on the sizes of the vertices of  $P$  are given:

**Theorem 58** *Let  $n \in \mathbb{N}$  and  $c \in \mathbb{Q}^n$ . Let  $P \subseteq \mathbb{R}^n$  be a rational polytope and let  $x_0 \in P$  be a vector in the interior of  $P$ . Let  $T$  be a positive integer such that  $\text{size}(x_0) \leq \log(T)$  and  $\text{size}(x) \leq \log(T)$  for all vertices  $x$  of  $P$ . Given  $n, c, x_0, T$  and a polynomial-time separation oracle for  $P$ , a vertex  $x^*$  of  $P$  attaining  $\max\{c^t x \mid x \in P\}$  can be found in time polynomial in  $n, \log(T)$  and  $\text{size}(c)$ .*

For a proof, we refer to Korte and Vygen [2018].

The other direction (from an optimization algorithm to a separation oracle) works as well:

**Theorem 59** *Let  $n \in \mathbb{N}$  and  $y \in \mathbb{Q}^n$ . Let  $P \subseteq \mathbb{R}^n$  be a rational polytope and let  $x_0 \in P$  be a vector in the interior of  $P$ . Let  $T$  be a positive integer such that  $\text{size}(x_0) \leq \log(T)$  and  $\text{size}(x) \leq \log(T)$  for all vertices  $x$  of  $P$ . Given  $n, y, x_0, T$  and an oracle which for given  $c \in \mathbb{Q}^n$  returns a vertex  $x^*$  of  $P$  attaining  $\max\{c^t x \mid x \in P\}$ , we can implement a separation oracle for  $P$  and  $y$  with running time polynomial in  $n, \log(T)$  and  $\text{size}(y)$ . If  $y \notin P$ , we can find with this running time a facet-defining inequality of  $P$  that is violated by  $y$ .*

For a proof, we again refer to Korte and Vygen [2018].



## 7 Interior Point Methods

The ELLIPSOID ALGORITHM gives a polynomial-time algorithm for solving linear programs but in practice it is typically much less efficient than the SIMPLEX ALGORITHM. In contrast, the algorithm that we will describe in this section is efficient both in theory and practice.

The term “interior point method” refers to several quite different algorithms. They all have in common that during the algorithm we always consider vectors in the interior of the polyhedron of feasible solutions (in contrast to the SIMPLEX ALGORITHM where we always have vectors on the border of the polyhedron). Here, we restrict ourselves to one variant and follow the description by Mehlhorn and Saxena [2015]. The first version of the algorithm has been proposed by Karmakar [1984].

We consider an LP  $\max\{c^t x \mid Ax \leq b\}$  in standard inequality form.

To simplify the notation, we write the slack variables  $s$  explicitly, so we consider the following problem:

$$\begin{aligned} & \max c^t x \\ \text{s.t.} \quad & Ax + s = b \\ & s \geq 0 \end{aligned} \tag{47}$$

We write its dual problem in standard form:

$$\begin{aligned} & \min b^t y \\ \text{s.t.} \quad & A^t y = c \\ & y \geq 0 \end{aligned} \tag{48}$$

In fact, what we will compute is a solution of this dual linear program.

In the following, we assume that the columns of  $A$  are linearly independent (otherwise we had redundant equations in the constraints of the dual LP) and that the number of rows is larger than the number of columns (otherwise we could simply solve the equation system for the dual and check if the solution is non-negative). These are the same assumptions that we had for the SIMPLEX ALGORITHM (but for the transposed matrix).

By complementary slackness, we have solved both problems to optimality when we have found a feasible solution  $x, s$  of the primal LP and a feasible solution  $y$  of the dual LP such that  $y^t s = 0$ . In other words, we want to find  $x, s$ , and  $y$  with:

$$\begin{aligned} Ax + s &= b \\ A^t y &= c \\ y^t s &= 0 \\ y &\geq 0 \\ s &\geq 0 \end{aligned} \tag{49}$$

Note that  $y^t s = 0$  is not a linear constraint. Without this constraint (i.e. for the system  $Ax + s = b, A^t y = c, y \geq 0, s \geq 0$ ), the term  $y^t s$  is exactly the difference between the

(dual) value of the dual solution  $y$  and the (primal) value of the primal solution  $x, s$  because  $b^t y - c^t x = x^t A^t y + s^t y - c^t x = x^t c + s^t y - c^t x = s^t y$ .

The system (49) has a solution only if both the primal and the dual linear program are feasible and bounded, so for the moment we assume that this is the case. In Section 7.1, we will see what to do to enforce these properties.

In the interior point methods, one generally considers vectors in the interior of the solution space. In the system (49), the only inequalities are  $y \geq 0$  and  $s \geq 0$ , so during the algorithm, we always have solutions  $x, s, y$  with  $y > 0$  and  $s > 0$ . We will replace the condition  $y^t s = 0$  by the condition  $\sigma^2 := \sum_{i=1}^m \left( \frac{y_i s_i}{\mu} - 1 \right)^2 \leq \frac{1}{4}$  for some number  $\mu > 0$ . During the iterations of the algorithm, we will decrease  $\mu$  more and more towards 0.

To summarize, during the algorithm, we have a number  $\mu > 0$  and vectors  $x, s, y$  meeting the following invariants

$$\begin{aligned} Ax + s &= b \\ A^t y &= c \\ \sum_{i=1}^m \left( \frac{y_i s_i}{\mu} - 1 \right)^2 &\leq \frac{1}{4} \\ y &> 0 \\ s &> 0 \end{aligned} \tag{50}$$

Now, the general strategy consists of three main parts:

- (I) Compute an initial solution of a modified version of (50) (Section 7.1).
- (II) Reduce  $\mu$  by a constant factor and adapt  $x, y$  and  $s$  to this new value of  $\mu$  such that we again get a solution of (50). Iterate this step until  $\mu$  is small enough (Section 7.2).
- (III) Compute an optimum solution of the dual LP (Section 7.3).

## 7.1 Modification of the LP and Computation of an Initial Solution

We will show how we can modify (50) to an equivalent problem that can be solved easily, provided that we are allowed to choose  $\mu$ . This modification will in particular make both the primal and the dual LP feasible. This is equivalent to the statement that one of them is feasible and bounded. We will show how to modify the dual LP (48) such that the modified version is feasible and bounded.

In a first step, we make the LP (48) bounded (in such a way that we do not change the problem if the given LP was bounded). By Theorem 42, we know that if (48) is feasible and bounded, then there is a  $W$  with  $W \in 2^{\Theta(m(\text{size}(A) + \text{size}(c)))}$  such that there is an optimum solution  $y = (y_1, \dots, y_m) \geq 0$  with  $y_i \leq W$  ( $i = 1, \dots, m$ ). So in this case there is a vector  $y \geq 0$  with  $\mathbb{1}^t y \leq mW$  and  $A^t y = c$ . Equivalently (after dividing everything by  $W$ ), we can ask for a vector  $y \geq 0$  with  $\mathbb{1}^t y \leq m$  and  $A^t y = \frac{1}{W}c$ . By relaxing the constraint  $\mathbb{1}^t y \leq m$  to  $\mathbb{1}^t y \leq m + 1$  and



by adding a slack variable  $y_{m+1} \geq 0$  this leads to the following LP which is equivalent to (48) provided that (48) is bounded:

$$\begin{aligned}
& \min b^t y \\
\text{s.t. } & A^t y = \frac{1}{W} c \\
& \mathbb{1}^t y + y_{m+1} = m + 1 \\
& y \geq 0 \\
& y_{m+1} \geq 0
\end{aligned} \tag{51}$$

In a second step, we will make the LP feasible. To this end, we add a new variable  $y_{m+2}$  such that setting all variables to 1 will get us a feasible solution. Let  $H$  be a constant (to be determined later). Then, we state the following LP:

$$\begin{aligned}
\min & b^t y + H y_{m+2} \\
\text{s.t. } & A^t y + \left(\frac{1}{W} c - A^t \mathbb{1}\right) y_{m+2} = \frac{1}{W} c \\
& \mathbb{1}^t y + y_{m+1} + y_{m+2} = m + 2 \\
& y \geq 0 \\
& y_{m+1} \geq 0 \\
& y_{m+2} \geq 0
\end{aligned} \tag{52}$$

The goal is to choose  $H$  that big that if this LP has a feasible solution with  $y_{m+2} = 0$  at all, then in *any* optimum solution  $y_{m+2} = 0$  will hold. In fact, by Corollary 43 we know that there is a constant  $l$  such that if there is an optimum solution of (52) with  $y_{m+2} > 0$ , then there is an optimum solution with  $y_{m+2} \geq 2^{-4ml(\text{size}(A)+\text{size}(c)+\text{size}(W))}$ . On the other hand,  $b^t y \leq \|b\|_1(m+2)$  in any feasible solution of (52), so if we set  $H = (\|b\|_1(m+2) + 1)2^{4ml(\text{size}(A)+\text{size}(c)+\text{size}(W))}$ , then we enforce that  $y_{m+2} = 0$  in any optimum solution (if a solution with  $y_{m+2} = 0$  exists).

The linear program (52) is obviously feasible and bounded. In addition, we can use an optimum solution of it, to check if the initial dual LP was feasible and bounded, and if this is the case, we can find an optimum solution of it: Let  $y_1, \dots, y_{m+2}$  be an optimum solution of (52). If  $y_{m+2} > 0$ , then we know that (51) has no feasible solution (otherwise there was a feasible solution of (52) with  $y_{m+2} = 0$  which is cheaper). Thus, the LP (48) has no feasible solution either. On the other hand, if  $y_{m+2} = 0$ , then the initial dual LP must be feasible. Assume that this is the case, then we still have to check if the initial dual LP was bounded. If  $y_{m+1} > 0$ , the initial dual program must be bounded. If  $y_{m+1} = 0$ , then the initial dual LP can be bounded or unbounded. To decide if it is bounded, we can replace  $c$  by the all-zero vector and first solve this new problem. Then, by Farkas' Lemma, the LP (48) is bounded if and only if the value of an optimum solution of the new problem is non-negative.

If we dualize the LP (52), we get the following LP (with variables  $x \in \mathbb{R}^n$ ,  $s \in \mathbb{R}^m$  and additional

variables  $x_{n+1}$ ,  $s_{m+1}$ , and  $s_{m+2}$ ):

$$\begin{aligned}
\max \quad & \frac{1}{W}c^t x + (m+2)x_{n+1} \\
& Ax + x_{n+1}\mathbb{1} + s & = b \\
\left(\frac{1}{W}c^t - \mathbb{1}^t A\right)x + x_{n+1} & + s_{m+2} & = H \\
& x_{n+1} + s_{m+1} & = 0 \\
& s & \geq 0 \\
& s_{m+1} & \geq 0 \\
& s_{m+2} & \geq 0
\end{aligned} \tag{53}$$

Instead of the primal-dual pair (47) and (48), we will consider the pair (52) and (53). Due to the modification, both LPs are feasible and bounded.

For the new pair of LPs we can easily find feasible solutions and a number  $\mu$  such that  $\sum_{i=1}^{m+2} \left(\frac{y_i s_i}{\mu} - 1\right)^2 \leq \frac{1}{4}$ : We set  $y_1 = y_2 = \dots = y_m = y_{m+1} = y_{m+2} = 1$  which is obviously feasible for (52). For (53), we set  $x_1 = x_2 = \dots = x_n = 0$ . Moreover, we choose  $s_{m+1} = \frac{\mu}{y_{m+1}} = \mu$  (where  $\mu$  itself is still to be determined). This leads to  $x_{n+1} = -\mu$ ,  $s_{m+2} = H + \mu$ , and  $s_i = b_i - x_{n+1} = b_i + \mu$  ( $i = 1, \dots, m$ ).

As a consequence of this choice, we get:

$$\begin{aligned}
\frac{y_i s_i}{\mu} - 1 & = \frac{b_i}{\mu} \quad i = 1, \dots, m \\
\frac{y_{m+1} s_{m+1}}{\mu} - 1 & = 0 \\
\frac{y_{m+2} s_{m+2}}{\mu} - 1 & = \frac{H}{\mu}
\end{aligned}$$

Therefore,

$$\sigma^2 = \sum_{i=1}^{m+2} \left(\frac{y_i s_i}{\mu} - 1\right)^2 = \frac{1}{\mu^2} \left(H^2 + \sum_{i=1}^m b_i^2\right).$$

Hence, by choosing  $\mu = 2\sqrt{H^2 + \sum_{i=1}^m b_i^2}$ , we enforce  $\sigma^2 \leq \frac{1}{4}$ . Moreover, since  $\mu > |b_i|$ , we have  $s_i = b_i + \mu > 0$  for  $i \in \{1, \dots, m\}$ .

So what did we get so far? We have replaced the primal-dual pair (47) and (48) by the pair (52) and (53) such that optimum solutions of these modified problems directly lead to a solution of the original problem. Moreover, the new primal-dual pair consists of two feasible and bounded problems.

We will write (52) as

$$\begin{aligned}
\min \quad & \tilde{b}^t y \\
\text{s.t.} \quad & \tilde{A}^t y = \tilde{c} \\
& y \geq 0
\end{aligned} \tag{54}$$

and (53) as

$$\begin{aligned} & \max \tilde{c}^t x \\ \text{s.t.} \quad & \tilde{A}x + s = \tilde{b} \\ & s \geq 0 \end{aligned} \tag{55}$$

so  $\tilde{A} \in \mathbb{R}^{(m+2) \times (n+1)}$ ,  $\tilde{b} \in \mathbb{R}^{m+1}$  and  $\tilde{c} \in \mathbb{R}^{n+1}$ .

Note that in these modified problems we have variables  $x \in \mathbb{R}^{n+1}$  and  $y, s \in \mathbb{R}^{m+2}$  (nevertheless we denote them by  $x, y, s$  as in (47) and (48)).

We have already found initial solutions  $\mu^{(0)}, x^{(0)}, y^{(0)}, s^{(0)}$  for the following system:

$$\begin{aligned} \tilde{A}x + s &= \tilde{b} \\ \tilde{A}^t y &= \tilde{c} \\ \sum_{i=1}^{m+2} \left( \frac{y_i s_i}{\mu} - 1 \right)^2 &\leq \frac{1}{4} \\ y &> 0 \\ s &> 0 \end{aligned} \tag{56}$$

## 7.2 Solutions for Reduced Values of $\mu$

In this section, we will describe a solution for the following problem: Given a solution  $\mu^{(k)}, x^{(k)}, y^{(k)}, s^{(k)}$  of (56) we want to compute a new solution  $\mu^{(k+1)}, x^{(k+1)}, y^{(k+1)}, s^{(k+1)}$  of (56) where  $\mu^{(k+1)} = (1 - \delta)\mu^{(k)}$  for some  $\delta$  that does not depend on the solution (to be determined later).

In a first version, we describe the step without considering the sizes of the numbers that occur during the computation. Afterwards, we will show how we can round intermediate solutions in such a way that the numbers can be written with a polynomial number of bits.

We write  $x^{(k+1)} = x^{(k)} + f$ ,  $y^{(k+1)} = y^{(k)} + g$ , and  $s^{(k+1)} = s^{(k)} + h$ . Think of the entries of  $f$ ,  $g$  and  $h$  as relatively small values. Assuming that  $\mu^{(k+1)}$  is fixed, we describe how to compute appropriate values for  $f$ ,  $g$  and  $h$ . The first two conditions of (56) lead to  $\tilde{A}f + h = 0$  and  $\tilde{A}^t g = 0$ . In addition we want to choose  $f$  and  $h$  such that  $(y_i^{(k)} + g_i)(s_i^{(k)} + h_i)$  is close to  $\mu^{(k+1)}$  ( $i = 1 \dots, m+2$ ). Since  $(y_i^{(k)} + g_i)(s_i^{(k)} + h_i) = y_i^{(k)} s_i^{(k)} + g_i s_i^{(k)} + y_i^{(k)} h_i + g_i h_i$  and the product  $g_i h_i$  is small (provided that  $g_i$  and  $h_i$  are small) we simply demand  $y_i^{(k)} s_i^{(k)} + g_i s_i^{(k)} + y_i^{(k)} h_i = \mu^{(k+1)}$  ( $i = 1 \dots, m+2$ ). Hence, we want to compute  $f$ ,  $g$  and  $h$  such that

$$\begin{aligned} \tilde{A}^t g &= 0 \\ \tilde{A}f + h &= 0 \\ s_i^{(k)} g_i + y_i^{(k)} h_i &= \mu^{(k+1)} - y_i^{(k)} s_i^{(k)} \quad i = 1, \dots, m+2 \end{aligned} \tag{57}$$

Note that  $y^{(k)}$  and  $s^{(k)}$  are constant in this context. In this formulation, we skipped the constraints that  $y^{(k+1)} > 0$  and  $s^{(k+1)} > 0$ . We will see what we can do to get positive values, anyway.

Let  $f$ ,  $g$  and  $h$  be a solution of (57). By construction, we have

$$(y^{(k)} + g)^t (s^{(k)} + h) = (m + 2)\mu^{(k+1)} + g^t h. \quad (58)$$

Furthermore the first and second constraint of (57) give

$$g^t h = -g^t \tilde{A} f = 0^t f = 0. \quad (59)$$

This implies

$$\begin{aligned} \tilde{b}^t y^{(k+1)} - \tilde{c}^t x^{(k+1)} &= \left( \tilde{A}(x^{(k)} + f) + (s^{(k)} + h) \right)^t (y^{(k)} + g) - \tilde{c}^t (x^{(k)} + f) \\ &= \left( \tilde{A}(x^{(k)} + f) \right)^t (y^{(k)} + g) + (m + 2)\mu^{(k+1)} - \tilde{c}^t (x^{(k)} + f) \\ &= (x^{(k)} + f)^t \tilde{A}^t y^{(k)} + (m + 2)\mu^{(k+1)} - \tilde{c}^t (x^{(k)} + f) \\ &= (m + 2)\mu^{(k+1)} \end{aligned} \quad (60)$$

**Lemma 60** *The system (57) has a unique solution.*

**Proof:** Let  $S$  be an  $(m + 2) \times (m + 2)$ -diagonal matrix with  $s_i^{(k)}$  as entry at position  $(i, i)$  and  $Y$  be an  $(m + 2) \times (m + 2)$ -diagonal matrix with  $y_i^{(k)}$  as entry at position  $(i, i)$ .

Then, the last condition of (57) is equivalent to

$$Sg + Yh = \mu^{(k+1)} \mathbb{1}_{m+2} - Sy^{(k)},$$

which is equivalent to

$$g + S^{-1}Yh = S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2} - y^{(k)}.$$

This implies

$$\tilde{A}^t g + \tilde{A}^t S^{-1}Yh = \tilde{A}^t S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2} - \tilde{A}^t y^{(k)}, \quad (61)$$

and hence

$$\tilde{A}^t S^{-1}Yh = \tilde{A}^t S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2} - \tilde{c}. \quad (62)$$

With  $h = -\tilde{A}f$  this leads to

$$-\tilde{A}^t S^{-1}Y\tilde{A}f = \tilde{A}^t S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2} - \tilde{c}.$$

However, the matrix  $\tilde{A}^t S^{-1}Y\tilde{A}$  is invertible, so  $f = (\tilde{A}^t S^{-1}Y\tilde{A})^{-1}(\tilde{c} - \tilde{A}^t S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2})$  is the unique solution of this last equation. In particular, if (57) has a solution, this is the only choice for  $f$ . By setting  $h = -\tilde{A}f$ , we fulfill the second constraint of (57). Finally, we set  $g = S^{-1}\mu^{(k+1)} \mathbb{1}_{m+2} - y^{(k)} - S^{-1}Yh$  (again the only choice) satisfying the third constraint of (57).

Since we have chosen  $g$  and  $h$  such that (61) and (62) are met, we also have  $\tilde{A}^t g = 0$ , so the solution satisfies the first condition of (57).  $\square$

We have  $\sigma^{(k)} = \sqrt{\sum_{i=1}^{m+2} \left( \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} - 1 \right)^2}$  and  $\sigma^{(k+1)} = \sqrt{\sum_{i=1}^{m+2} \left( \frac{y_i^{(k+1)} s_i^{(k+1)}}{\mu^{(k+1)}} - 1 \right)^2} = \sqrt{\sum_{i=1}^{m+2} \left( \frac{g_i h_i}{\mu^{(k+1)}} \right)^2}$ .

It remains to show that  $y^{(k+1)} > 0$  and  $s^{(k+1)} > 0$  and  $\sigma^{(k+1)} \leq \frac{1}{2}$ .

We first show that for an appropriate choice of  $\mu^{(k+1)}$  we get  $\sigma^{(k+1)} \leq \frac{1}{2}$ .

**Lemma 61** (a) For  $i = 1, \dots, m+2$  we have  $\frac{\mu^{(k)}}{y_i^{(k)} s_i^{(k)}} \leq \frac{1}{1-\sigma^{(k)}}$ .

$$(b) \sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| \leq \sigma^{(k)} \sqrt{m+2}.$$

**Proof:**

(a) We have  $(\sigma^{(k)})^2 = \sum_{i=1}^{m+2} \left( \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} - 1 \right)^2$ , so  $\left( \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} - 1 \right)^2 \leq (\sigma^{(k)})^2$  which implies  $\left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| \leq \sigma^{(k)}$  and  $\frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \geq 1 - \sigma^{(k)}$  for  $i = 1, \dots, m+2$ . This proves the claim.

(b) The statement is simply a special case of the Cauchy-Schwarz inequality that can be proved as follows:

$$\begin{aligned} & (\sigma^{(k)})^2 (m+2) - \left( \sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| \right)^2 \\ &= (m+2) \sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right|^2 - \left( \sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| \right)^2 \\ &= (m+1) \sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right|^2 - 2 \sum_{i=1}^{m+2} \sum_{j=i+1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| \cdot \left| 1 - \frac{y_j^{(k)} s_j^{(k)}}{\mu^{(k)}} \right| \\ &= \sum_{i=1}^{m+2} \sum_{j=i+1}^{m+2} \left( \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right| - \left| 1 - \frac{y_j^{(k)} s_j^{(k)}}{\mu^{(k)}} \right| \right)^2 \\ &\geq 0 \end{aligned}$$

This proves (b). □

**Lemma 62** If  $\delta = \frac{1}{8\sqrt{m+2}}$  (i.e.  $\mu^{(k+1)} = (1 - \frac{1}{8\sqrt{m+2}})\mu^{(k)}$ ) then  $\sigma^{(k+1)} < \frac{1}{2}$ .

**Proof:** Let  $G_i := g_i \sqrt{\frac{s_i^{(k)}}{y_i^{(k)} \mu^{(k+1)}}$  and  $H_i := h_i \sqrt{\frac{y_i^{(k)}}{s_i^{(k)} \mu^{(k+1)}}$  (for  $i \in \{1, \dots, m+2\}$ ).

$$\begin{aligned}
\sigma^{(k+1)} &= \sqrt{\sum_{i=1}^{m+2} \left( \frac{g_i h_i}{\mu^{(k+1)}} \right)^2} = \sqrt{\sum_{i=1}^{m+2} (G_i H_i)^2} \\
&= \sqrt{\frac{1}{4} \left( \sum_{i=1}^{m+2} (G_i^2 + H_i^2)^2 - \sum_{i=1}^{m+2} (G_i^2 - H_i^2)^2 \right)} \\
&\leq \frac{1}{2} \sqrt{\sum_{i=1}^{m+2} (G_i^2 + H_i^2)^2} \leq \frac{1}{2} \sum_{i=1}^{m+2} (G_i^2 + H_i^2) \\
&\stackrel{g^t h=0}{=} \frac{1}{2} \sum_{i=1}^{m+2} (G_i + H_i)^2 = \frac{1}{2} \sum_{i=1}^{m+2} \frac{1}{y_i^{(k)} s_i^{(k)} \mu^{(k+1)}} \underbrace{(g_i s_i^{(k)} + h_i y_i^{(k)})^2}_{=\mu^{(k+1)} - y_i^{(k)} s_i^{(k)}} \\
&= \frac{1}{2} \sum_{i=1}^{m+2} \frac{(\mu^{(k)})^2}{y_i^{(k)} s_i^{(k)} \mu^{(k+1)}} \left( \frac{\mu^{(k+1)}}{\mu^{(k)}} - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2 \\
&= \frac{1}{2} \sum_{i=1}^{m+2} \underbrace{\frac{\mu^{(k)}}{y_i^{(k)} s_i^{(k)}}}_{\leq \frac{1}{1-\sigma^{(k)}}} \frac{1}{1-\delta} \left( -\delta + 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2 \\
&\leq \frac{1}{2(1-\delta)(1-\sigma^{(k)})} \left( (m+2)\delta^2 - 2\delta \sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right) + \sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2 \right) \\
&\leq \frac{1}{2(1-\delta)(1-\sigma^{(k)})} \left( (m+2)\delta^2 + 2\delta \underbrace{\sum_{i=1}^{m+2} \left| 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right|}_{\leq \sigma^{(k)} \sqrt{m+2}} + \underbrace{\sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2}_{=(\sigma^{(k)})^2} \right) \\
&\leq \frac{1}{2(1-\delta)(1-\sigma^{(k)})} \left( (m+2)\delta^2 + 2\delta \sigma^{(k)} \sqrt{m+2} + (\sigma^{(k)})^2 \right) \\
&= \frac{1}{2(1-\delta)(1-\sigma^{(k)})} \left( \sqrt{m+2} \delta + \sigma^{(k)} \right)^2 \\
&\stackrel{\sigma^{(k)} \leq \frac{1}{2}}{\leq} \frac{1}{1-\delta} \left( \sqrt{m+2} \delta + \frac{1}{2} \right)^2 = \frac{8\sqrt{m+2}}{8\sqrt{m+2}-1} \left( \frac{1}{8} + \frac{1}{2} \right)^2 \\
&\leq \frac{1}{2}.
\end{aligned}$$

□

**Lemma 68** We have  $y^{(k+1)} > 0$  and  $s^{(k+1)} > 0$ .

**Proof:** Claim: We have  $y_i^{(k+1)} s_i^{(k+1)} > 0$  for  $i = 1, \dots, m+2$ .

Proof of the Claim:

Assume that  $y_j^{(k+1)} s_j^{(k+1)} \leq 0$  for a  $j \in \{1, \dots, m+2\}$ . Then,

$$(\sigma^{(k+1)})^2 = \sum_{i=1}^{m+2} \left( \frac{y_i^{(k+1)} s_i^{(k+1)}}{\mu^{(k+1)}} - 1 \right)^2 \geq \left( \frac{y_j^{(k+1)} s_j^{(k+1)}}{\mu^{(k+1)}} - 1 \right)^2 \geq 1,$$

which is a contradiction to the previous lemma. This proves the claim.

Thus if  $y_i^{(k+1)} \leq 0$ , then  $s_i^{(k+1)} \leq 0$  and vice versa. Assume that  $y_i^{(k+1)} = y_i^{(k)} + g_i \leq 0$  and  $s_i^{(k+1)} = s_i^{(k)} + h_i \leq 0$ . This implies (because  $s_i^{(k)} > 0$  and  $y_i^{(k)} > 0$ ) that

$$\underbrace{s_i^{(k)}(y_i^{(k)} + g_i) + y_i^{(k)}(s_i^{(k)} + h_i)}_{=s_i^{(k)}y_i^{(k)} + \mu^{(k+1)}} \leq 0$$

which is a contradiction to the fact that  $s_i^{(k)}$ ,  $y_i^{(k)}$ , and  $\mu^{(k+1)}$  are positive. □

### Rounding the intermediate solution

When computing the modification vectors  $f$ ,  $g$  and  $h$  according to (57), we have to avoid that the number of bits needed to store the numbers increases too much in each iteration. We can do this in the following way: Instead of the exact values of  $y_i^{(k)}$  and  $s_i^{(k)}$ , we solve the system (57) with respect to rounded values  $\tilde{y}_i^{(k)}$  and  $\tilde{s}_i^{(k)}$ . We do this in such a way that they remain positive and such that  $|\frac{\tilde{y}_i^{(k)} \tilde{s}_i^{(k)}}{\mu^{(k)}} - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}}| < \epsilon$  for some  $\epsilon$  with  $0 < \epsilon < \frac{1}{m+2} \frac{1}{300}$ . By restricting the solution space to a polytope we can assume that a polynomial number of bits is sufficient to store these rounded numbers  $\tilde{y}_i^{(k)}$  and  $\tilde{s}_i^{(k)}$ .

Then, we get  $\sum_{i=1}^{m+2} \left( 1 - \frac{\tilde{y}_i^{(k)} \tilde{s}_i^{(k)}}{\mu^{(k)}} \right)^2 \leq \sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2 + \sum_{i=1}^{m+2} 2 \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right) \epsilon + \sum_{i=1}^{m+2} \epsilon^2 \leq \sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2 + \frac{1}{100}$  Thus, if we can bound  $\sum_{i=1}^{m+2} \left( 1 - \frac{y_i^{(k)} s_i^{(k)}}{\mu^{(k)}} \right)^2$  by, say, 0.49 instead of 0.5, we get  $\sum_{i=1}^{m+2} \left( 1 - \frac{\tilde{y}_i^{(k)} \tilde{s}_i^{(k)}}{\mu^{(k)}} \right)^2 \leq \frac{1}{2}$ . For the initial solution, this is easy (simply increase the initial value  $\mu^{(0)}$  slightly). For the intermediate step, this is also not an issue because in the proof of Lemma 62, we easily get in the very last inequality even 0.49 as the upper bound.

## 7.3 Finding an Optimum Solution

We will describe a way to find an optimum solution of the dual LP (54).

For the remainder of the chapter, we use the following notation: Let  $y^*$  be an optimum solution of (54) and  $x^*, s^*$  an optimum solution of (55). By Corollary 43, we can assume that all positive entries of  $y^*$  and  $s^*$  have a value of at least  $\eta$  for some  $\eta = 2^{-\Theta(\text{size}(\tilde{A})+\text{size}(\tilde{b})+\text{size}(\tilde{c}))}$ .

**Lemma 63** *Let  $\mu, x, y, s$  be a solution of (56). Let  $i \in \{1, \dots, m+2\}$ . Then:*

(a) *If  $y_i < \frac{\eta}{4(m+2)}$ , then  $y_i^* = 0$ .*

(b) *If  $s_i < \frac{\eta}{4(m+2)}$ , then  $s_i^* = 0$ .*

**Proof:** By the condition  $\sum_{i=1}^{m+2} \left( \frac{y_i s_i}{\mu} - 1 \right)^2 \leq \frac{1}{4}$ , we get

$$\frac{\mu}{2} \leq y_i s_i \leq \frac{3\mu}{2} < 2\mu$$

for all  $i \in \{1, \dots, m+2\}$ . Therefore,  $s^t y = \sum_{i=1}^{m+2} y_i s_i \leq 2(m+2)\mu$ .

(a) Since  $y^*$  is an optimum and  $y$  a feasible solution of the dual LP, we have  $\tilde{b}^t y \geq \tilde{b}^t y^*$  and thus

$$s^t y = \tilde{b}^t y - x^t \tilde{A}^t y = \tilde{b}^t y - \tilde{c}^t x \geq \tilde{b}^t y^* - \tilde{c}^t x = \tilde{b}^t y^* - x^t \tilde{A}^t y^* = s^t y^*.$$

Let  $i \in \{1, \dots, m+2\}$  with  $y_i < \frac{\eta}{4(m+2)}$ . We have

$$s_i \geq \frac{\mu}{2y_i} > \frac{2(m+2)\mu}{\eta} \geq \frac{s^t y}{\eta}.$$

Assume that  $y_i^* > 0$ , so  $y_i^* \geq \eta$ . This implies

$$s^t y^* \geq s_i y_i^* > \frac{s^t y}{\eta} \cdot \eta = s^t y \geq s^t y^*,$$

which is a contradiction. Therefore,  $y_i^* = 0$ .

(b) The case is very similar to part (a): Since  $x^*, s^*$  is an optimum and  $x, s$  a feasible solution of the primal LP, we have  $\tilde{c}^t x \leq \tilde{c}^t x^*$  and thus

$$s^t y = \tilde{b}^t y - x^t \tilde{A}^t y = \tilde{b}^t y - \tilde{c}^t x \geq \tilde{b}^t y - \tilde{c}^t x^* = \tilde{b}^t y - y^t \tilde{A} x^* = y^t s^*.$$

Let  $i \in \{1, \dots, m+2\}$  with  $s_i < \frac{\eta}{4(m+2)}$ . We have

$$y_i \geq \frac{\mu}{2s_i} > \frac{2(m+2)\mu}{\eta} \geq \frac{s^t y}{\eta}.$$

Assume that  $s_i^* > 0$ , so  $s_i^* \geq \eta$ . This implies

$$y^t s^* \geq s_i^* y_i > \eta \cdot \frac{s^t y}{\eta} = s^t y \geq y^t s^*,$$

which is again a contradiction. Therefore,  $s_i^* = 0$ . □



There are several ways to find an optimum solution. Before we describe a method to round an interior point directly to an optimum solution, we will present a simpler but less efficient method: We choose  $k$  big enough such that  $\mu^{(k)} < \frac{\eta^2}{32(m+2)^2}$ . Then, for each  $i \in \{1, \dots, m+2\}$ , we have  $y_i^{(k)} < \frac{\eta}{4(m+2)}$  or  $s_i^{(k)} < \frac{\eta}{4(m+2)}$ . Let  $\bar{A}^t y = \bar{c}$  be the subsystem of  $\tilde{A}^t y = \tilde{c}$  consisting of the rows with indices  $i$  for which  $s_i^{(k)} < \frac{\eta}{4(m+2)}$ , so  $s_i^* = 0$ . For all other rows, we know that  $y_i^* = 0$ , so we can ignore them when computing an optimum solution for the dual LP. If  $\bar{A}^t y = \bar{c}$  has only one solution, we compute it and get an optimum solution of the modified dual LP (52) (provided that the result is non-negative). Otherwise, we check if  $y_{i_0}^{(k)} < \frac{\eta}{4(m+2)}$  for some  $i_0 \in \{1, \dots, m\}$ . In this case we know that if the initial dual LP has an optimal solution, then there is one with  $y_{i_0} = 0$ . Hence we can start the whole process again but now without the variable  $y_{i_0}$ , without the row of  $A$  with index  $i_0$  and without the entry of  $b$  with index  $i_0$ . Hence we have reduced the instance size, so this method will terminate after at most  $m$  iterations.

What can we do if there is no  $i \in \{1, \dots, m\}$  with  $y_i^{(k)} < \frac{\eta}{4(m+2)}$ ? To handle this case, we first make sure that the system  $\tilde{A}x = \tilde{b}$  does not have a feasible solution. If it has a feasible solution (which can be checked by Gaussian Elimination), we modify  $\tilde{b}$  slightly to a vector  $b^*$  such that  $\tilde{A}x = b^*$  has no feasible solution. To this end choose  $n$  linearly independent rows of  $A$ . These rows will define the solution of  $\tilde{A}x = \tilde{b}$ . Then, any modification of  $b$  outside these rows will make the system  $\tilde{A}x = \tilde{b}$  infeasible. We simply add an  $\epsilon > 0$  to one of these entries of  $b$ . If  $\epsilon$  is small enough, then an optimum solution of the dual LP with respect to  $b^*$  will still be an optimum solution of the original dual LP. To see that we can write  $\epsilon$  with a polynomial number of bits, observe that the absolute value of the difference between the costs of two basic solutions of an LP is either 0 or can be bounded from below by some value  $2^{-L}$  where  $L$  is polynomial in the input size. This follows from the fact that any basic solution can be written with a polynomial number of bits. Thus, the same is true for any difference  $u$  of two basic solutions and for the scalar product  $\tilde{b}^t u$ . Hence,  $\tilde{b}^t u$  is either zero or its absolute value is at least  $2^{-L}$ . This implies that we can choose  $\epsilon$  in such a way that it can be written with polynomially many bits and that no suboptimal solution can become optimal by the modification.

Now assume that the initial dual LP is bounded and feasible. Then, we can compute optimum solutions  $x^*, y^*, s^*$  of the modified LPs (52) and (53) by expanding optimum solutions of the initial primal and dual problems in a canonical way. In particular, we will set  $x_{n+1}$  to 0. Then  $Ax^* + s^* = b^*$  but  $Ax = b^*$  has no feasible solution. Hence, there must be an  $i_0 \in \{1, \dots, m\}$  with  $s_{i_0}^* > 0$ , so  $y_{i_0}^{(k)} < \frac{\eta}{4(m+2)}$  and  $y_{i_0}^* = 0$ . Again, we get rid of at least one dual variable and can restart the whole procedure on a smaller instance.

Now, we describe how we can avoid iterating the whole process:

Consider again the two problems (54) and (55). Theorem 13 implies that we can partition the index set  $\{1, \dots, m+2\}$  of the dual variables into  $\{1, \dots, m+2\} = B \dot{\cup} N$  such that for  $i \in B$  there is an optimum dual solution  $y^*$  with  $y_i^* > 0$  and for  $i \in N$  there is an optimum primal solution  $x^*, s^*$  with  $s_i^* > 0$ . Any optimum solution can be written as convex combination of basic solutions. Hence, in Lemma 63 for any  $i \in \{1, \dots, m+2\}$  we can either have  $y_i < \frac{\eta}{4(m+2)}$  or  $s_i < \frac{\eta}{4(m+2)}$  but not both. Now we choose  $k$  big enough such that  $\mu^{(k)} < \frac{\eta^2}{32(m+2)^2 \Delta}$  for

some  $\Delta \geq 1$  that will be determined later. Then, for each  $i \in \{1, \dots, m+2\}$ , exactly one of the inequalities  $y_i < \frac{\eta}{4(m+2)\Delta}$  and  $s_i < \frac{\eta}{4(m+2)\Delta}$  holds. Therefore, we can find the partitioning  $\{1, \dots, m+2\} = B \dot{\cup} N$ . In particular, we have  $y_i \geq \frac{\eta}{4(m+2)}$  for each  $i \in B$  and  $y_i < \frac{\eta}{4(m+2)\Delta}$  for each  $i \in N$ .

Let  $A_B$  be the submatrix of  $\tilde{A}$  consisting of the rows with indices in  $B$ , and  $A_N$  be the submatrix of  $\tilde{A}$  consisting of the remaining rows. By  $y_B^{(k)}, y_N^{(k)}, b_B, b_N$  we denote the corresponding subvectors of vectors  $y^{(k)}$  and  $b$ . As in the description of the **SIMPLEX ALGORITHM**, the entries of e.g.  $y_B^{(k)}$  are not necessarily indexed from 1 to  $|B|$  but their index set is the set  $B \subseteq \{1, \dots, m+2\}$ . We can assume that  $A_B$  has full column rank.

In the following, the vector norm is the Euclidean norm  $\|\cdot\|_2$  and the matrix norm is the norm induced by the Euclidean norm.

**Theorem 64** *Set  $\Delta = \max\{\sqrt{(m+2)}\|A_B(A_B^t A_B)^{-1} A_N^t\|, 1\}$ . Let  $k$  be big enough such that  $\mu^{(k)} < \frac{\eta^2}{32(m+2)^2\Delta}$ . Let  $Y_B$  be a diagonal matrix whose rows and columns are indexed with  $B$  such that the entry at position  $(i, i)$  is  $y_i^{(k)}$ . Define*

$$d_y := Y_B A_B (A_B^t (Y_B)^2 A_B)^{-1} A_N^t y_N^{(k)}$$

and  $\tilde{y}_B = Y_B d_y + y_B^{(k)}$ . Then:

(a)  $A_B^t \tilde{y}_B = \tilde{c}$ .

(b)  $\|d_y\| < 1$ .

(c) The vector  $\tilde{y} \in \mathbb{R}^{m+2}$  which arises from  $\tilde{y}_B$  by adding zeros for the entries with index in  $N$  is an optimum dual solution.

**Proof:**

(a) We have  $A_B^t (Y_B d_y + y_B^{(k)}) = A_N^t y_N^{(k)} + A_B^t y_B^{(k)} = \tilde{c}$ .

(b)

$$\begin{aligned} \|d_y\| &= \|Y_B A_B (A_B^t (Y_B)^2 A_B)^{-1} A_N^t y_N^{(k)}\| \\ &= \|Y_B A_B (A_B^t (Y_B)^2 A_B)^{-1} \underbrace{A_B^t Y_B Y_B^{-1} A_B (A_B^t A_B)^{-1}}_{=I_n} A_N^t y_N^{(k)}\| \\ &= \underbrace{\|Y_B A_B (A_B^t (Y_B)^2 A_B)^{-1} A_B^t Y_B\|}_{=1} \cdot \|Y_B^{-1} A_B (A_B^t A_B)^{-1} A_N^t y_N^{(k)}\| \\ &\leq \underbrace{\|Y_B^{-1}\|}_{\leq \frac{4(m+2)}{\eta}} \cdot \underbrace{\|A_B (A_B^t A_B)^{-1} A_N^t\|}_{\leq \frac{\Delta}{\sqrt{m+2}}} \cdot \underbrace{\|y_N^{(k)}\|}_{< \frac{\eta\sqrt{m+2}}{4(m+2)\Delta}} \\ &\leq 1. \end{aligned}$$

- (c) By (a), we have  $\tilde{A}^t \tilde{y} = \tilde{c}$ , and by (b), we know that  $\tilde{y}_B > 0$ , so we have  $\tilde{y} \geq 0$ . Hence  $\tilde{y}$  is a feasible dual solution. Moreover, we know that there is a feasible primal solution in which the slack variables  $s_i$  are zero for  $i \in B$ . Hence, by complementary slackness,  $\tilde{y}$  is an optimum dual solution.  $\square$

**Theorem 65** *Given a feasible and bounded linear program  $\min\{b^t y \mid y^t A = c^t, y \geq 0\}$  with  $A \in \mathbb{Q}^{m \times n}$ ,  $b \in \mathbb{Q}^m$ , and  $c \in \mathbb{Q}^n$ , the INTERIOR POINT METHOD computes an optimum solution in polynomial time. Moreover, the algorithm decides correctly, if a linear program is feasible or bounded.  $\square$*

## 7.4 Further remarks

Interior point methods can be motivated as follows: When we consider the primal LP

$$\begin{aligned} & \max c^t x \\ \text{s.t. } & Ax + s = b \\ & s \geq 0 \end{aligned} \tag{63}$$

we may enforce an interior point as a solution by replacing the objective function  $c^t x$  by  $c^t x + \mu \sum_{i=1}^m \ln(s_i)$  for some parameter  $\mu > 0$ . If there is any solution in the interior, then in any optimum solution of the modified problem, we have  $s_i > 0$  for all  $i \in \{1, \dots, m\}$ . And the larger  $\mu$  is, the more we are enforced to find a solution away from border of the solution space. That is why  $c^t x + \mu \sum_{i=1}^m \ln(s_i)$  is called **barrier function** and the modified problem is called **barrier problem**. Then, one can relax the constraints  $Ax + s = b$  by some **Lagrangian relaxation** and consider the following function, called **Lagrange function**

$$L(x, s, y) = c^t x + \mu \sum_{i=1}^m \ln(s_i) + y^t (b - Ax - s).$$

For any fixed vector  $y$  the maximum value  $\tilde{L}(y) = \max\{L(x, s, y) \mid x \in \mathbb{R}^n, s \in \mathbb{R}^m\}$  is an upper bound on  $\max\{c^t x + \mu \sum_{i=1}^m \ln(s_i) \mid Ax + s = b\}$ , and hence one asks for  $\min\{\tilde{L}(y) \mid y \in \mathbb{R}^m\}$ . In this setting, one can show that  $c^t x + \mu \sum_{i=1}^m \ln(s_i)$  is maximum if all partial derivatives of  $L(x, s, y)$  are zero, i.e. all the following terms must be zero:

$$\begin{aligned} \frac{\partial L(x, s, y)}{\partial x_j} &= c_j - \sum_{i=1}^m y_i a_{ij} && \text{for } j \in \{1, \dots, n\} \\ \frac{\partial L(x, s, y)}{\partial s_i} &= \frac{\mu}{s_i} - y_i && \text{for } i \in \{1, \dots, m\} \\ \frac{\partial L(x, s, y)}{\partial y_i} &= b_i - \sum_{j=1}^n a_{ij} x_j - s_i && \text{for } i \in \{1, \dots, m\} \end{aligned}$$

This leads to  $y^t A = c$ ,  $y_i s_j = \mu$  (for  $j \in \{1, \dots, n\}$ ), and  $Ax + s = b$ . Note that we do not get explicit sign constraints on  $y$  but if we request  $s$  to be positive, then  $y$  will be positive. For a decreasing sequence of  $\mu^{(k)}$  values, the sequence of corresponding vectors  $x^{(k)}$ ,  $s^{(k)}$  and  $y^{(k)}$  satisfying  $(y^{(k)})^t A = c$ ,  $y_i^{(k)} s_j^{(k)} = \mu^{(k)}$  (for  $j \in \{1, \dots, n\}$ ), and  $Ax^{(k)} + s^{(k)} = b^{(k)}$  is called a **central path**. However, note that also the sequence of solutions we constructed in the first three subsection with only  $y_i^{(k)} s_j^{(k)} \approx \mu^{(k)}$  is called central path.

## 8 Integer Linear Programming

Imposing integrality constraints on all or some variables of a linear program allows to model many new conditions that could not be described by linear constraints. For example, even if we only consider **BINARY LINEAR PROGRAMS** (i.e. all integrality constraints are of the type  $x \in \{0, 1\}$ ) we can easily model the following conditions for variables  $x, y$ :

- “ $(x \geq a \text{ or } y \geq b) \text{ and } x, y \geq 0$ ” for some  $a, b > 0$ .
- “ $x \in \{s_1, \dots, s_k\}$ ” for a set  $\{s_1, \dots, s_k\}$  of real numbers.

On the other hand, we have already seen that there are NP-hard optimization problems that can be modeled as (mixed) integer linear programs. Hence, we cannot hope for polynomial-time algorithms to solve general ILPs.

### 8.1 Integral Polyhedra

**Definition 17** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a polyhedron. Then, we define  $P_I := \text{conv}\{x \in \mathbb{Z}^n \mid Ax \leq b\}$  as the **integer hull** of  $P$ .

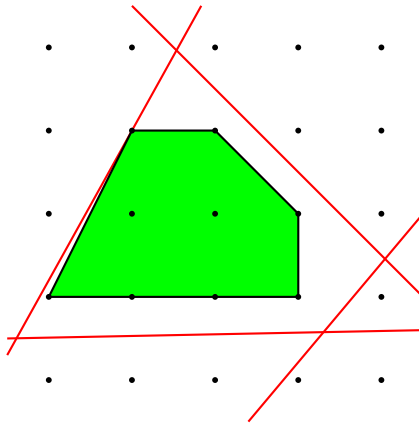


Fig. 8: A polyhedron  $P$  (given by the red hyperplanes) and its integer hull  $P_I$  (green). The black dots indicate the integral vectors.

#### Observations:

- For a rational polyhedral cone (i.e. a cone  $C = \{x \in \mathbb{R}^n \mid Ax \leq 0\}$  with  $A \in \mathbb{Q}^{m \times n}$ ), we have  $C_I = C$  (because a polyhedral cone is rational if and only if it is generated by a finite number of integral vectors).

- $P_I$  is not necessarily a polyhedron.
- If  $P$  is a polytope, then  $P_I$  is a polyhedron.

**Theorem 66** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a rational polyhedron, i.e.  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . Then,  $P_I$  is a polyhedron.

**Proof:** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . By Theorem 31, we can write  $P = \text{conv}(V) + \text{cone}(E)$  for two finite sets  $V, E \subseteq \mathbb{R}^n$ . Moreover, the proof of Theorem 31 also shows that we can assume that the elements of  $V$  and  $E$  are rational vectors. Hence, we can even assume that  $E = \{y_1, \dots, y_s\}$  where  $y_i$  are integral vectors ( $i = 1, \dots, s$ ). Define  $B := \{\sum_{i=1}^s \lambda_i y_i \mid 0 \leq \lambda_i \leq 1 \text{ for } i \in \{1, \dots, s\}\}$ .

Claim:  $P_I = (\text{conv}(V) + B)_I + \text{cone}(E)$ .

Proof of the claim: “ $P_I \subseteq (\text{conv}(V) + B)_I + \text{cone}(E)$ ”:

Let  $p$  be an integral vector of  $P$ . Then,  $p = q + c$  for some  $q \in \text{conv}(V)$  and some  $c \in \text{cone}(E)$ . We can write  $c = \sum_{i=1}^s \mu_i y_i$  with  $\mu_i \geq 0$  for  $i \in \{1, \dots, s\}$ . Therefore  $c = \sum_{i=1}^s \mu_i y_i = \underbrace{\sum_{i=1}^s (\mu_i - \lfloor \mu_i \rfloor) y_i}_{\in B} + \underbrace{\sum_{i=1}^s \lfloor \mu_i \rfloor y_i}_{\in (\text{cone}(E) \cap \mathbb{Z}^n)}$ , so we can write  $c = b + c'$  with  $b \in B$  and  $c' \in \text{cone}(E) \cap \mathbb{Z}^n$ .

Thus,  $p = (q + b) + c'$ . We have  $q + b \in \text{conv}(V) + B$ . And  $q + b = p - c'$ , so  $q + b$  is integral. Hence,  $q + b \in (\text{conv}(V) + B)_I$ , and therefore  $p \in (\text{conv}(V) + B)_I + \text{cone}(E)$ .

“ $P_I \supseteq (\text{conv}(V) + B)_I + \text{cone}(E)$ ”:

We have

$$(\text{conv}(V) + B)_I + \text{cone}(E) \subseteq P_I + \text{cone}(E) = P_I + (\text{cone}(E))_I \subseteq (P + \text{cone}(E))_I = P_I.$$

This proves the claim.

The claim implies the statement of the theorem because  $\text{conv}(V) + B$  is a polytope, so  $(\text{conv}(V) + B)_I$  is also a polytope. This shows that  $P_I$  can be written as the Minkowski sum of two polyhedra. However, by an earlier exercise, the Minkowski sum of two polyhedra is again a polyhedron.  $\square$

In particular, we get the (somewhat surprising) consequence that one can solve integer linear programs by solving linear programs. The problem is that the polyhedron  $P_I$  may not have a simple description even if there is one for  $P$ . For example, Rubin [1970] has shown that for any  $k$  there are rational polyhedra  $P \subseteq \mathbb{R}^2$  with only 3 facets such that  $P_I$  has more than  $k$  facets. Moreover, Bárány, Howe, and Lovász [1992] gave an example showing that there are rational polyhedra  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\} \subseteq \mathbb{R}^n$  such that  $P_I$  has  $\Omega(\phi^{n-1})$  vertices, where  $\phi = \text{size}(A) + \text{size}(b)$ .

**Definition 18** A polyhedron  $P$  is called **integral** if  $P = P_I$ .

**Proposition 67** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$  such that  $P_I \neq \emptyset$ . Let  $c \in \mathbb{R}^n$  be a vector. Then,  $\max\{c^t x \mid x \in P\}$  is bounded if and only if  $\max\{c^t x \mid x \in P_I\}$  is bounded.

**Proof:** “ $\Rightarrow$ ” trivial.

“ $\Leftarrow$ ” Assume that  $\max\{c^t x \mid x \in P\}$  is unbounded. Then, the dual LP must be infeasible, so there is no vector  $y$  with  $y^t A = c$  and  $y \geq 0$ . By Farkas’ Lemma (Theorem 6), this means that there is a vector  $z$  with  $c^t z < 0$  and  $Az \geq 0$ . Thus, the LP  $\min\{c^t x \mid Ax \geq 0, -\mathbb{1} \leq x \leq \mathbb{1}\}$  is feasible and has an optimum solution with negative value. By Proposition 42, there is a rational optimum solution  $x^*$ . By multiplying  $x^*$  by an appropriate integer, we get an integral vector  $w$  with  $Aw \geq 0$  and  $c^t w < 0$ . Hence, for any  $v \in P_I$  and  $k \in \mathbb{N}$  we have  $v - kw \in P_I$ . Therefore,  $\max\{c^t x \mid x \in P_I\}$  is unbounded.  $\square$

## 8.2 Integral Solutions of Equation Systems

In this section, our goal is to find a certificate that a given system of equations does *not* have any integral solution (which will be the result of Corollary 69).

**Definition 19** An  $m \times n$ -matrix  $A$  is in **Hermite normal form** if it can be written as  $A = [B \ 0]$  where  $B$  is a nonsingular lower triangular non-negative matrix such that each row of  $B$  has a unique maximum entry and this maximum entry is on the diagonal.

The following operations on matrices are called **elementary unimodular column operations**:

- Exchange two columns.
- Multiply a column by  $-1$ .
- Add an integral multiple of one column to another column.

**Theorem 68** Each matrix  $A \in \mathbb{Q}^{m \times n}$  of rank  $m$  can be transformed into a matrix in Hermite normal form by a series of elementary unimodular column operations.

**Proof:** We may assume that  $A$  is integral. Assume that we have already transformed  $A$  into a matrix  $\begin{bmatrix} F & 0 \\ G & H \end{bmatrix}$  where  $F$  is a lower triangular matrix with positive diagonal. Let  $h_{11}, \dots, h_{1k}$  be the first row of  $H$ . Apply elementary unimodular column operations to  $H$  such that all  $h_{1j}$  are non-negative and such that  $\sum_{j=1}^k h_{1j}$  is as small as possible. We may assume that  $h_{11} \geq h_{12} \geq \dots \geq h_{1k}$ . Then,  $h_{11} > 0$  because  $A$  has rank  $m$ . Moreover,  $h_{1j} = 0$  for  $j \in \{2, \dots, k\}$  because otherwise subtracting  $h_{1j}$  from  $h_{11}$  would reduce  $\sum_{j=1}^k h_{1j}$ . Hence, we have obtained a larger lower triangular matrix  $F'$ .

We iterate this step and end up with a matrix  $[B \ 0]$  where  $B$  is a lower triangular matrix with positive diagonal. Denote the entries of  $B$  be  $b_{ij}$  ( $i = 1, \dots, m, j = 1, \dots, m$ ). Finally, we perform for  $i = 2, \dots, m$  the following steps: For  $j = 1, \dots, i - 1$  add an integer multiple of the  $i$ -th column of  $B$  to the  $j$ -th column of  $B$  such that the  $b_{ij}$  is non-negative and less than  $b_{ii}$ .  $\square$

**Corollary 69** *Let  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . Then,  $Ax = b$  has an integral solution  $x$  if and only if  $b^t y$  is integral for each  $y \in \mathbb{Q}^m$  for which  $A^t y$  is integral.*

**Proof:** “ $\Rightarrow$ ” If  $x$  and  $y^t A$  are integral vectors and  $Ax = b$ , then  $y^t Ax = y^t b$  is also integral.

“ $\Leftarrow$ ” Assume that  $b^t y$  is integral for each  $y \in \mathbb{Q}^m$  for which  $A^t y$  is integral. Then,  $Ax = b$  must have a (fractional) solution, since otherwise, by Farkas’ Lemma (Corollary 7), there would be a vector  $y \in \mathbb{Q}^m$  with  $y^t A = 0$  and  $y^t b = -\frac{1}{2}$ . Thus, we may assume that the rows of  $A$  are linearly independent, so  $A$  has rank  $m$ .

It is easy to check the statement to be proved holds for  $A$  if and only if it holds for any matrix  $\tilde{A}$  where  $\tilde{A}$  arises from  $A$  by applying an elementary unimodular column operation. Hence, we can assume that  $A$  is in Hermite normal form  $[B \ 0]$ . Thus  $B^{-1}[B \ 0] = [I_m \ 0]$  is an integral matrix. Therefore by our assumption (applied to the rows of  $B^{-1}$ ),  $B^{-1}b$  is an integral vector. Since  $[B \ 0] \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix} = b$ , the vector  $x := \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$  is an integral solution for  $[B \ 0]x = b$ .  $\square$



### 8.3 TDI Systems

**Theorem 70** *Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Q}^m$ . Then, the following statements are equivalent:*

- (a)  *$P$  is integral*
- (b) *Each face of  $P$  contains at least one integral vector.*
- (c) *Each minimal face of  $P$  contains at least one integral vector.*
- (d) *Each supporting hyperplane of  $P$  contains at least one integral vector.*
- (e) *Each rational supporting hyperplane of  $P$  contains at least one integral vector.*
- (f)  *$\max\{c^t x \mid x \in P\}$  is attained by an integral vector for each  $c$  for which the maximum is finite.*
- (g)  *$\max\{c^t x \mid x \in P\}$  is an integer for each integral vector  $c$  for which the maximum is finite.*

**Proof:** The following implications are obvious: “(b)  $\Leftrightarrow$  (c)”, “(b)  $\Rightarrow$  (d)”, “(d)  $\Rightarrow$  (e)”, and “(f)  $\Rightarrow$  (g)”

“(a)  $\Rightarrow$  (b):” Assume that  $P$  is integral. Let  $F = P \cap H$  be a face of  $P$  where  $H = \{x \in \mathbb{R}^n \mid c^t x = \delta\}$  is a supporting hyperplane of  $P$ . Then, any  $z \in F$  is a convex combination of integral vectors  $v_1, \dots, v_k$  of  $P$ . If  $v_i \in P \setminus F$  (so  $c^t v_i < \delta$ ) for an  $i \in \{1, \dots, k\}$ , then (since  $c^t x = \delta$ ) there must be a  $j \in \{1, \dots, k\}$  with  $c^t v_j > \delta$ , which is a contradiction to  $v_j \in P$ . Thus, all  $v_i$  must be in  $F$ , so in particular  $F$  contains an integral vector.

“(c)  $\Rightarrow$  (f):” Follows from Corollary 19.

“(f)  $\Rightarrow$  (a):” Assume that (f) holds but  $P \neq P_I$ . Then, there is an  $x^* \in P \setminus P_I$ . By Theorem 66,  $P_I$  is a polyhedron, so there is an inequality  $a^t x \leq \beta$  that is valid for  $P_I$  but not for  $x^*$ , so  $a^t x^* > \beta$ . This is a contradiction to (f) because  $\max\{a^t x \mid x \in P\}$  is finite (by Proposition 67) but is not attained by any integral vector.

So far, we have proved that (a),(b),(c), and (f) are equivalent.

“(e)  $\Rightarrow$  (c):” We may assume that  $A$  and  $b$  are integral. Let  $F = \{x \in \mathbb{R}^n \mid A'x = b'\}$  be a minimal face of  $P$  (where  $A'x \leq b'$  is a subsystem of  $Ax \leq b$ ). If there is no integral vector  $x$  with  $A'x = b'$ , then, by Corollary 69, there must be a rational vector such that  $c := (A')^t y$  is integral while  $\delta := y^t b'$  is not an integer. Moreover, we may assume that all entries of  $y$  are positive (otherwise we add an appropriate integral vector to  $y$ ). Since  $c$  is integral but  $\delta$  is not integral, the rational hyperplane  $H := \{x \in \mathbb{R}^n \mid c^t x = \delta\}$  does not contain any integral vector.

We will show that  $H \cap P = F$  which implies that  $H$  is a supporting hyperplane. By construction,

we have  $F \subseteq H$ , so we have to show that  $H \cap P \subseteq F$ . Let  $x \in H \cap P$ . Then,  $y^t A'x = c^t x = \delta = y^t b'$ , so  $y^t(A'x - b') = 0$ . Thus, since all components of  $y$  are positive,  $A'x = b'$ , so  $x \in F$ .

Now, we know that (a),(b),(c),(d),(e), and (f) are equivalent.

“(g)  $\Rightarrow$  (e):” Let  $H = \{x \in \mathbb{R}^n \mid c^t x = \delta\}$  be a rational supporting hyperplane of  $P$ , so  $\max\{c^t x \mid x \in P\} = \delta$ . Assume that  $H$  does not contain any integral vector. Then, by Corollary 69, there is a positive number  $\gamma$  for which  $\gamma c$  is integral but  $\gamma \delta$  is not integral. Then  $\max\{(\gamma c)^t x \mid x \in P\} = \gamma \max\{c^t x \mid x \in P\} = \gamma \delta \notin \mathbb{Z}$ , so the statement of (g) is false.

Since “(f)  $\Rightarrow$  (g)” is trivial, this shows the equivalence of all statements.  $\square$

Note that this Theorem implies that for any rational polyhedron  $P \subseteq \mathbb{R}^n$  with  $P = P_I$  and any rational vector  $c$  there is a polynomial-time algorithm computing a vector  $x \in P \cap \mathbb{Z}^n$  maximizing  $c^t x$  over  $P \cap \mathbb{Z}^n$ , provided that there is an optimum solution. To this end, we only have to compute an integral element of a minimal face  $F$  consisting of optimum solutions only (for finding  $F$  we can apply the ELLIPSOID METHOD). This can be done by computing an integral solution of an equation system, which is possible in polynomial time by the method described in the proof of Corollary 69.

Moreover, by the equivalence of (f) and (g), the existence of an integral solution can be deduced from the integrality of the solution value. This motivates the following definition:

**Definition 20** *A system of inequalities  $Ax \leq b$  is called **totally dual integrals (TDI-system)**, if the LP  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  has an integral optimum solution for each integral vector  $c$  for which the LP is feasible and bounded.*

Note that total dual integrality is in fact a property of the system of inequalities, not just of the polyhedron that is defined by them. For example the systems

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

and

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

define the same polyhedron. But it is easy to check that the first system of inequalities is TDI while the second one is not TDI.

**Theorem 71** *Let  $A \in \mathbb{Q}^{m \times n}$  and  $b \in \mathbb{Z}^m$  such that  $Ax \leq b$  is totally dual integral. Then, the polyhedron  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  is integral.*

**Proof:** If  $Ax \leq b$  is TDI, then by definition  $\min\{b^t y \mid A^t y = c, y \geq 0\}$  is an integer for each integral vector  $c$  for which the minimum is finite. By duality, this implies that  $\max\{c^t x \mid Ax \leq b\}$

is an integer for each integral vector  $c$  for which the maximum is finite. Thus, by the implication “(g)  $\Rightarrow$  (a)” of Theorem 70,  $P$  is integral.  $\square$

**Proposition 72** *If  $Ax \leq b$  is a TDI-system, and  $a^t x \leq \beta$  is valid for any  $x \in \mathbb{R}^n$  with  $Ax \leq b$ , then the system  $Ax \leq b, a^t x \leq \beta$  is also totally dual integral.*

**Proof:** Exercise.  $\square$

Hence, if a system  $Ax \leq b$  is not TDI, then no proper subsystem  $A'x \leq b'$  with  $\{x \in \mathbb{R}^n \mid Ax \leq b\} = \{x \in \mathbb{R}^n \mid A'x \leq b'\}$  can be TDI. We call a system  $Ax \leq b$  **minimally TDI** if it is TDI but no proper subsystem of  $Ax \leq b$  defining the same polyhedron is TDI.

**Proposition 73** *If  $Ax \leq b, a^t x \leq \beta$  is a TDI-system with integral vector  $a$ , then  $Ax \leq b, a^t x = \beta$  is also a TDI-system.*

**Proof:** Let  $c$  be an integral vector for which

$$\begin{aligned} & \max\{c^t x \mid Ax \leq b, a^t x = \beta\} \\ &= \min\{b^t y + \beta(\lambda - \mu) \mid y \geq 0, \lambda, \mu \geq 0, A^t y + (\lambda - \mu)a = c\} \end{aligned} \quad (64)$$

is finite. Let  $x^*, y^*, \lambda^*, \mu^*$  be optimum primal and dual solutions. Set  $\tilde{c} := c + \lceil \mu^* \rceil a$ . Then,

$$\begin{aligned} & \max\{\tilde{c}^t x \mid Ax \leq b, a^t x \leq \beta\} \\ &= \min\{b^t y + \beta\lambda \mid y \geq 0, \lambda \geq 0, A^t y + \lambda a = \tilde{c}\} \end{aligned} \quad (65)$$

is finite because  $x^*$  is feasible for the maximum and  $y^*$  and  $\lambda^* + \lceil \mu^* \rceil - \mu^*$  are feasible for the minimum.

Since  $Ax \leq b, a^t x \leq \beta$  is a TDI-system, the minimum in equation (65) has an integer optimum solution  $\tilde{y}, \tilde{\lambda}$ . Then,  $y := \tilde{y}, \lambda := \tilde{\lambda}, \mu := \lceil \mu^* \rceil$  is an integer optimum solution for the minimum in (64): it is obviously feasible, and its cost is:

$$b^t \tilde{y} + \beta(\tilde{\lambda} - \lceil \mu^* \rceil) = b^t \tilde{y} + \beta\tilde{\lambda} - \beta\lceil \mu^* \rceil \leq b^t y^* + \beta(\lambda^* + \lceil \mu^* \rceil - \mu^*) - \beta\lceil \mu^* \rceil = b^t y^* + \beta(\lambda^* - \mu^*).$$

The inequality follows from the fact that  $y^*, \lambda^* + \lceil \mu^* \rceil - \mu^*$  is feasible for the minimum in (65) and  $\tilde{y}, \tilde{\lambda}$  is an optimum solution for the minimum in (65). Hence, the minimum in (64) has an integral optimum solution, so  $Ax \leq b, a^t x = \beta$  is TDI.  $\square$

**Definition 21** *A finite set of vectors  $\{v_1, \dots, v_t\}$  is a **Hilbert basis** if each integral vector in  $\text{cone}(\{v_1, \dots, v_t\})$  is a non-negative integral combination of  $v_1, \dots, v_t$ .*

**Example:** The set of the standard unit vectors together with their negatives is a Hilbert basis of the cone  $\mathbb{R}^n$ .

**Theorem 74** *Every rational polyhedral cone is generated by an integral Hilbert basis.*

**Proof:** Let  $C$  be a rational polyhedral cone.  $C$  is generated by some rational vectors  $b_1, \dots, b_k$ , and we can assume with loss of generality that these vectors are integral. Let  $H$  consist of all integral vectors in

$$P = \left\{ \sum_{i=1}^k \lambda_i b_i \mid 0 \leq \lambda_i \leq 1 \text{ for } i \in \{1, \dots, k\} \right\}.$$

Obviously  $H$  is a finite set. We claim that  $H$  is a Hilbert basis generating  $C$ . As  $\{b_1, \dots, b_k\} \subseteq H \subseteq C$ , the cone  $C$  is generated by  $H$ . To see that  $H$  forms a Hilbert basis, let  $b$  be an integral vector in  $C$ . Since  $b_1, \dots, b_k$  generate  $C$ , there are nonnegative numbers  $\mu_1, \dots, \mu_k$  with  $b = \sum_{i=1}^k \mu_i b_i$ , so

$$b = \left( \sum_{i=1}^k \lfloor \mu_i \rfloor b_i \right) + \sum_{i=1}^k (\mu_i - \lfloor \mu_i \rfloor) b_i.$$

Then, the vector

$$b - \left( \sum_{i=1}^k \lfloor \mu_i \rfloor b_i \right) = \sum_{i=1}^k (\mu_i - \lfloor \mu_i \rfloor) b_i$$

is integral and an element of  $P$ . Thus (since  $\{b_1, \dots, b_k\} \subseteq H$ ),  $b$  can be written as a non-negative integral combination of the elements of  $H$ . This shows that  $H$  is a Hilbert basis.  $\square$

**Notation:** For a system of inequalities  $Ax \leq b$  and a face  $F$  of  $\{x \in \mathbb{R}^n \mid Ax \leq b\}$ , we call a row of  $A$  **active**, if the corresponding inequality in  $Ax \leq b$  is satisfied with equality for all  $x \in F$ .

**Theorem 75** *A feasible rational system of inequalities  $Ax \leq b$  is TDI if and only if for each minimal face  $F$  of  $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$ , the rows that are active in  $F$  form a Hilbert basis.*

**Proof:** “ $\Rightarrow$ .” Suppose that  $Ax \leq b$  is TDI. Let  $F$  be a minimal face of  $P$  and let  $a_1, \dots, a_t$  be the rows of  $A$  that are active for  $F$ . We have to show that  $\{a_1, \dots, a_t\}$  is a Hilbert basis. Let  $c$  be an integral vector in  $\text{cone}(\{a_1, \dots, a_t\})$ . We have to write  $c$  as an integral non-negative combination of  $a_1, \dots, a_t$ . The maximum in the LP-duality equation

$$\max\{c^t x \mid Ax \leq b\} = \min\{b^t y \mid A^t y = c, y \geq 0\} \quad (66)$$

is attained by every vector  $x$  in  $F$ . Since  $Ax \leq b$  is TDI, the dual problem has an integral optimum solution  $y$ . By complementary slackness, the entries of  $y$  at positions corresponding to rows that are not active in  $F$  are 0. Thus,  $c$  is an integral non-negative combination of  $a_1, \dots, a_t$ .

“ $\Leftarrow$ ” Assume that for each minimal face  $F$  of  $P$ , the rows that are active in  $F$  form a Hilbert basis. Let  $c$  be an integral vector for which the optima in (66) are finite. We have to show that the minimum is attained by an integral vector. Let  $F$  be a minimal face of  $P$  such that each vector in  $F$  attains the maximum in the duality equation. Let  $a_1, \dots, a_t$  be rows of  $A$  that are active in  $F$ . Then, by complementary slackness,  $c \in \text{cone}(\{a_1, \dots, a_t\})$ . Since  $a_1, \dots, a_t$  form a Hilbert basis, we can write  $c = \sum_{i=1}^t \lambda_i a_i$  for certain non-negative integral numbers  $\lambda_1, \dots, \lambda_t$ . We can extend  $(\lambda_1, \dots, \lambda_t)$  with zero-components to a vector  $y \in \mathbb{Z}^m$  with  $y \geq 0$ ,  $A^t y = c$  and  $b^t y = x^t A^t y = c^t x$  for all  $x \in F$ . In other words,  $y$  is an integral optimum solution of the dual LP.  $\square$

**Theorem 76** *The rational system of inequalities  $Ax \leq 0$  is TDI if and only if the rows of  $A$  form a Hilbert basis.*

**Proof:** Follows from the previous Theorem with  $b = 0$  (note that in the unique minimal face of  $\{x \in \mathbb{R}^n \mid Ax \leq 0\}$  all rows of  $A$  are active).  $\square$

**Theorem 77** *(Giles and Pulleyblank [1979]) For each rational polyhedron  $P \subseteq \mathbb{R}^n$  there exists a rational TDI-system  $Ax \leq b$  with  $A \in \mathbb{Z}^{m \times n}$  and  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . The vector  $b$  can be chosen to be integral if and only if  $P$  is integral.*

**Proof:** We can assume w.l.o.g. that  $P \neq \emptyset$ . For each minimal face  $F$  of  $P$ , we define

$$C_F := \{c \in \mathbb{R}^n \mid c^t z = \max\{c^t x \mid x \in P\} \text{ for all } z \in F\}.$$

Then,  $C_F$  is a polyhedral cone. To see this, assume that  $P = \{\tilde{A}x \leq \tilde{b}\}$  is some description of  $P$ . Then  $C_F$  is generated by the rows of  $\tilde{A}$  that are active in  $F$ .

Let  $F$  be a minimal face, and let  $a_1, \dots, a_t$  be an integral Hilbert basis generating  $C_F$ . Choose  $x_0 \in F$ , and define  $\beta_i := a_i^t x_0$  for  $i = 1, \dots, t$ . Then,  $\beta_i = \max\{a_i^t x \mid x \in P\}$  ( $i = 1, \dots, t$ ). Let  $\mathcal{S}_F$  be the system  $a_1^t x \leq \beta_1, \dots, a_t^t x \leq \beta_t$ . All inequalities in  $\mathcal{S}_F$  are valid for  $P$ . Let  $Ax \leq b$  be the union of the systems  $\mathcal{S}_F$  over all minimal faces  $F$  of  $P$ . Then,  $P \subseteq \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . On the other hand, if  $x^* \in \mathbb{R}^n \setminus P$ , then there is a supporting hyperplane of  $P$  separating  $x^*$  from  $P$ , and this supporting hyperplane touches  $P$  in a minimal face, so there is an inequality in  $Ax \leq b$  that is violated by  $x^*$ . Hence,  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . Moreover, by Theorem 75,  $Ax \leq b$  is TDI.

If  $P$  is integral, then all the  $\beta_i$  can be chosen to be integral because we can choose the vectors  $x_0 \in F$  as integral vectors. On the other hand, if  $b$  is integral, then by Theorem 71,  $P$  is integral.  $\square$

In the primal-dual  $\max\{c^t x \mid Ax \leq b\} = \min\{b^t y \mid A^t y = c, y \geq 0\}$  we know (by the SIMPLEX ALGORITHM) that if both optima are finite, the minimization problem has an optimum solution  $y$  with at most  $\text{rank}(A)$  non-zero entries. If we ask for an optimum *integral* solution (with

$Ax \leq b$  TDI and  $b$  integral), this is not necessarily the case: see  $A = \begin{pmatrix} 2 \\ -3 \end{pmatrix}$ ,  $b = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$  and  $c = (1)$ . Nevertheless, for full-dimensional solution spaces, we get the following bound on the number of non-zero entries:

**Theorem 78** *Let  $Ax \leq b$  be a TDI-system with  $A \in \mathbb{Z}^{m \times n}$  such that  $\dim(\{x \in \mathbb{R}^n \mid Ax \leq b\}) = n$ . Let  $c$  be an integral vector for which the optima in*

$$\max\{c^t x \mid Ax \leq b\} = \min\{b^t y \mid A^t y = c, y \geq 0\}$$

*are finite. Then, the minimization problem has an integral optimum solution  $y$  with at most  $2r - 1$  positive components where  $r := \text{rank}(A)$ .*

**Proof: Claim:** Let  $\{a_1, \dots, a_t\} \subseteq \mathbb{Z}^n$  be a Hilbert basis such that  $C := \text{cone}(\{a_1, \dots, a_t\})$  is a pointed  $k$ -dimensional polyhedral cone. Then, any integral vector  $c$  in  $C$  is a nonnegative integral combination of at most  $2k - 1$  vectors in  $a_1, \dots, a_t$

**Proof of the Claim:** Let  $\lambda_1, \dots, \lambda_t$  attain

$$\max \left\{ \sum_{i=1}^t \lambda_i \mid \lambda_1, \dots, \lambda_t \geq 0; c = \sum_{i=1}^t \lambda_i a_i \right\} \quad (67)$$

Since  $C$  is pointed, the maximum is finite (check that the dual LP  $\min\{c^t y \mid y^t a_i \geq 1 \text{ for } i \in \{1, \dots, t\}\}$  is feasible). We can assume that at most  $k$  of the  $\lambda_i$  are non-zero. Define

$$c' := c - \sum_{i=1}^t \lfloor \lambda_i \rfloor a_i = \sum_{i=1}^t (\lambda_i - \lfloor \lambda_i \rfloor) a_i.$$

Then,  $c'$  is an integral vector in  $C$ , so we can write it as  $c' = \sum_{i=1}^t \mu_i a_i$  for some integral numbers  $\mu_1, \dots, \mu_t \geq 0$ . Since  $\lambda_1, \dots, \lambda_t$  was an optimum solution of (67) and  $\mu_1 + \lfloor \lambda_1 \rfloor, \dots, \mu_t + \lfloor \lambda_t \rfloor$  is a feasible solution, we have  $\sum_{i=1}^t \mu_i + \sum_{i=1}^t \lfloor \lambda_i \rfloor \leq \sum_{i=1}^t \lambda_i$ , so

$$\sum_{i=1}^t \mu_i \leq \sum_{i=1}^t \lambda_i - \sum_{i=1}^t \lfloor \lambda_i \rfloor < k$$

because at most  $k$  of the  $\lambda_i$  are non-zero. Thus, at most  $k - 1$  of the  $\mu_i$  are non-zero. Therefore, the decomposition

$$c = \sum_{i=1}^t (\lfloor \lambda_i \rfloor + \mu_i) a_i$$

has at most  $2k - 1$  non-zero components. This proves the claim.

The claim implies the statement of the theorem. To see this, first note that if  $P := \{x \in \mathbb{R}^n \mid Ax \leq b\}$  is full-dimensional, then a cone generated by rows of  $A$  that are active in a minimal face  $F$  of  $P$  must be pointed. Otherwise such a cone would contain a pair of vectors  $v$  and  $-v$ .

Thus there would be inequalities  $v^t x \leq \beta_1$  and  $-v^t x \leq \beta_2$  (for some numbers  $\beta_1, \beta_2$ ) that can be written as non-negative combinations of inequalities in  $Ax \leq b$  corresponding to rows of  $A$  that are active in  $F$ . For  $x \in F$  we have  $v^t x = \beta_1$  and  $-v^t x = \beta_2$ , so this would imply  $\beta_1 = -\beta_2$  and  $P$  would be contained in  $\{x \in \mathbb{R}^n \mid v^t x = \beta_1\}$ , which is a contradiction to the assumption that  $P$  is full-dimensional.

By Theorem 75, the rows that are active for a minimal face consisting of optimum solutions of  $\max\{c^t x \mid Ax \leq b\}$  form a Hilbert basis (because  $Ax \leq b$  is TDI).  $\square$

## 8.4 Total Unimodularity

In this section, we want to identify integral matrices  $A$  such that  $Ax \leq b, x \geq 0$  is TDI for any vector  $b$ . It will turn out that these are exactly the totally unimodular matrices (see Corollary 83).

**Definition 22** An  $m \times n$ -matrix  $A$  with rank  $m$  is called **unimodular** if  $A \in \mathbb{Z}^{m \times n}$  and for all regular  $m \times m$ -submatrices  $B$  of  $A$ , we have  $\det(B) \in \{-1, 1\}$ .

In particular, a regular square matrix is unimodular if and only if it is integral and its determinant is  $-1$  or  $1$ . Moreover, by Cramer's rule, the inverse of any unimodular square matrix is an integral matrix.

**Exercise:** Check that any series of elementary unimodular column operations, applied to a matrix  $A$  (see Chapter 8.2), can be performed by multiplying  $A$  from the right by an appropriate regular unimodular square matrix.

**Definition 23** A matrix  $A$  is called **totally unimodular** (TU) if every subdeterminant of  $A$  (i.e. every determinant of quadratic submatrices of  $A$ ) is  $0, -1$  or  $1$ .

In particular, all entries of totally unimodular matrices must be  $0, -1$  or  $1$ .

**Observation:** A matrix  $A$  is totally unimodular if and only if  $[I_m \ A]$  is unimodular.

**Theorem 79** Let  $A$  be a totally unimodular matrix, and let  $b$  be an integral vector. Then, the polyhedron  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  is integral.

**Proof:** Let  $F$  be a minimal face of  $P$ . We will show that  $F$  contains an integral vector. By the implication "(c)  $\Rightarrow$  (a)" of Theorem 70 this is sufficient to prove that  $P$  is integral.

By Proposition 22, we can write the minimal face as  $F = \{x \in \mathbb{R}^n \mid A'x = b'\}$  where  $A'x \leq b'$  is a subsystem of  $Ax \leq b$ . We can assume that  $A'$  has full row rank. By permuting coordinates,

we can write  $A' = [UV]$  for some matrix  $U$  with  $\det(U) \in \{-1, 1\}$ . Thus  $x := \begin{pmatrix} U^{-1}b' \\ 0 \end{pmatrix}$  is an integral vector in  $F$ .  $\square$

**Theorem 80** *Let  $A \in \mathbb{Z}^{m \times n}$  be a matrix with rank  $m$ . Then  $A$  is unimodular if and only if for each integral vector  $b$  the polyhedron  $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$  is integral.*

**Proof:** “ $\Rightarrow$ ” Assume that  $A$  is unimodular, and let  $b$  be an integral vector. Let  $x'$  be a vertex of  $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$ . This means that there are  $n$  linearly independent constraints in the system  $Ax \leq b, -Ax \leq -b, -I_n x \leq 0$  that are satisfied by  $x'$  with equality. Thus, the columns of  $A$  corresponding to non-zero entries of  $x'$  are linearly independent. This set of columns can be extended to a regular  $m \times m$ -submatrix  $B$  of  $A$ . Then, the restriction of  $x'$  to coordinates corresponding to  $B$  is  $B^{-1}b$ . This is integral (because  $\det(B) \in \{-1, 1\}$ ). The other entries of  $x'$  are zero, so  $x'$  is integral.

“ $\Leftarrow$ ” Suppose that  $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$  is integral for every integral vector  $b$ . Let  $B$  be a regular  $m \times m$ -submatrix of  $A$ . We have to show that  $\det(B) \in \{-1, 1\}$ . To this end, it is sufficient to show that  $B^{-1}u$  is integral for every integral vector  $u$  (by Cramer’s rule). So let  $u$  be an integral vector. Then, there is an integral vector  $y$  such that  $z := y + B^{-1}u \geq 0$ . Then,  $b := Bz$  is integral. Let  $z'$  be a vector with  $Az' = Bz = b$  that arises from  $z$  by adding zero-entries. Then,  $z'$  is a feasible (i.e. non-negative) basic solution of  $Ax = b$ , so it is a vertex of  $\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}$ . Therefore  $z'$  is integral, which also shows that  $z$  is integral. This implies that  $B^{-1}u = z - y$  is integral.  $\square$

**Theorem 81** (Hoffman and Kruskal [1956]) *Let  $A$  be an integral matrix. Then  $A$  is totally unimodular if and only if for each integral vector  $b$  the polyhedron  $\{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$  is integral.*

**Proof:** The matrix  $A$  is totally unimodular if and only if  $[I_m A]$  is unimodular. Let  $b$  be an integral vector. Then, the vertices of  $\{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$  are integral if and only if the vertices of  $\{z \in \mathbb{R}^{m+n} \mid [I_m A]z = b, z \geq 0\}$  are integral. Thus, the statement follows from Theorem 80.  $\square$

**Corollary 82** *An integral matrix  $A$  is totally unimodular if and only if for all integral vectors  $b$  and  $c$  optimum values for both sides of the duality equation*

$$\max\{c^t x \mid Ax \leq b, x \geq 0\} = \min\{b^t y \mid A^t y \geq c, y \geq 0\}$$

*are attained by integral vectors (if they are finite).*

**Proof:** Follows directly from Hoffmans and Kruskal’s Theorem (Theorem 81) using the fact



that a matrix is totally unimodular if and only if its transposed matrix is totally unimodular.  $\square$

**Corollary 83** *An integral matrix  $A$  is totally unimodular if and only if the system  $Ax \leq b, x \geq 0$  is TDI for each vector  $b$ .*

**Proof:** “ $\Rightarrow$ .” If  $A$  is totally unimodular, then also  $A^t$  is totally unimodular. Thus, by Theorem 81,  $\min\{b^t y \mid A^t y \geq c, y \geq 0\}$  is attained by an integral vector for each vector  $b$  and each integral vector  $c$  for which the minimum is finite. This implies that the system  $Ax \leq b, x \geq 0$  is TDI for each vector  $b$ .

“ $\Leftarrow$ .” Suppose that  $Ax \leq b, x \geq 0$  is TDI for each vector  $b$ . By Theorem 71 this implies that the polyhedron  $\{x \in \mathbb{R}^n \mid Ax \leq b, x \geq 0\}$  is integral for each integral vector  $b$ . By Theorem 81, this means that  $A$  is totally unimodular.  $\square$

The following theorem provides as a certificate to show that a matrix is totally unimodular.

**Theorem 84** (Ghoulia-Houri [1962]) *A matrix  $A = (a_{ij})_{\substack{i=1,\dots,m \\ j=1,\dots,n}} \in \mathbb{Z}^{m \times n}$  is totally unimodular if and only if for each set  $R \subseteq \{1, \dots, n\}$  there are sets  $R_1$  and  $R_2$  with  $R = R_1 \dot{\cup} R_2$  such that for each  $i \in \{1, \dots, m\}$ :*

$$\sum_{j \in R_1} a_{ij} - \sum_{j \in R_2} a_{ij} \in \{-1, 0, 1\}.$$

**Proof:** “ $\Rightarrow$ .” Let  $A$  be totally unimodular and  $R \subseteq \{1, \dots, n\}$ . Let  $d \in \{0, 1\}^n$  be the characteristic vector for  $R$ , i.e.

$$d_r = \begin{cases} 1 & \text{for } r \in R \\ 0 & \text{for } r \in \{1, \dots, n\} \setminus R \end{cases}$$

Since  $A$  is totally unimodular, also the matrix  $\begin{pmatrix} A \\ -A \\ I_n \end{pmatrix}$  is also totally unimodular. Thus, the polytope

$$P := \left\{ x \in \mathbb{R}^n \mid Ax \leq \left\lceil \frac{1}{2} Ad \right\rceil, Ax \geq \left\lfloor \frac{1}{2} Ad \right\rfloor, x \leq d, x \geq 0 \right\}$$

is integral. It contains the vector  $\frac{1}{2}d$ , so it is non-empty. Let  $z$  be an integral vertex of  $P$ . Then, for any  $i \in \{1, \dots, m\}$ , we have  $\sum_{j=1}^n a_{ij} z_j \leq \left\lfloor \frac{1}{2} \sum_{j=1}^n a_{ij} d_j \right\rfloor \leq \frac{1}{2} + \frac{1}{2} \sum_{j=1}^n a_{ij} d_j$  and  $\sum_{j=1}^n a_{ij} z_j \geq \left\lceil \frac{1}{2} \sum_{j=1}^n a_{ij} d_j \right\rceil \geq -\frac{1}{2} + \frac{1}{2} \sum_{j=1}^n a_{ij} d_j$ , so  $-1 \leq \sum_{j=1}^n a_{ij} (d_j - 2z_j) \leq 1$ .

Define  $R_1 := \{r \in R \mid z_r = 0\}$  and  $R_2 := \{r \in R \mid z_r = 1\}$ . For  $i \in \{1, \dots, m\}$ , this yields

$$\sum_{j \in R_1} a_{ij} - \sum_{j \in R_2} a_{ij} = \sum_{j=1}^n a_{ij}(d_j - 2z_j) \in \{-1, 0, 1\}$$

“ $\Leftarrow$ ” Assume that for each  $R \subseteq \{1, \dots, n\}$  there are sets  $R_1, R_2 \subseteq R$  with  $R = R_1 \dot{\cup} R_2$  as described in the theorem. We show by induction in  $k$  that every  $k \times k$ -submatrix of  $A$  has determinant  $-1, 0$ , or  $1$ . For  $k = 1$  this follows from the criterion for  $|R| = 1$ .

Let  $k > 1$ . Let  $B = (b_{ij})_{i,j \in \{1, \dots, k\}}$  a submatrix of  $A$ . We can assume that  $B$  is non-singular because otherwise its determinant is  $0$ .

By Cramer’s rule, each entry of  $B^{-1}$  is  $\frac{\det(B')}{\det(B)}$  where  $B'$  arises from  $B$  by replacing a column by a unit vector. By the induction hypothesis  $\det(B') \in \{-1, 0, 1\}$ . Hence, all entries of the matrix  $B^* := (\det(B))B^{-1}$  are in  $\{-1, 0, 1\}$ .

Let  $b^*$  be the first column of  $B^*$ . Then,  $Bb^* = \det(B)e_1$  where  $e_1$  is the first unit vector. We define  $R := \{j \in \{1, \dots, k\} \mid b_j^* \neq 0\}$ . For  $i \in \{2, \dots, k\}$ , we have  $0 = (Bb^*)_i = \sum_{j \in R} b_{ij}b_j^*$ , so  $|\{j \in R \mid b_{ij} \neq 0\}|$  is even.

Let  $R = R_1 \dot{\cup} R_2$  such that  $\sum_{j \in R_1} b_{ij} - \sum_{j \in R_2} b_{ij} \in \{-1, 0, 1\}$  for all  $i \in \{1, \dots, k\}$ . Thus, for  $i \in \{2, \dots, k\}$ , we have (since  $|\{j \in R \mid b_{ij} \neq 0\}|$  is even):  $\sum_{j \in R_1} b_{ij} - \sum_{j \in R_2} b_{ij} = 0$ . If we also had  $\sum_{j \in R_1} b_{1j} - \sum_{j \in R_2} b_{1j} = 0$ , then the columns of  $B$  would not be linearly independent. Hence,  $\sum_{j \in R_1} b_{1j} - \sum_{j \in R_2} b_{1j} \in \{-1, 1\}$  and thus,  $Bx \in \{e_1, -e_1\}$  where the vector  $x \in \{-1, 0, 1\}^k$  is defined by

$$x_j = \begin{cases} 1 & \text{for } j \in R_1 \\ -1 & \text{for } j \in R_2 \\ 0 & \text{for } j \in \{1, \dots, k\} \setminus R \end{cases}$$

Therefore,  $b^* = \det(B)B^{-1}e_1 \in \{\det(B)x, -\det(B)x\}$ . But both  $b^*$  and  $x$  are non-zero vectors with entries  $-1, 0, 1$  only, so we can conclude that  $\det(B) \in \{-1, 1\}$ .  $\square$

This result allows us to prove total unimodularity for some quite important matrices: The **incidence matrix** of an undirected graph  $G$  is the matrix  $A_G = (a_{v,e})_{\substack{v \in V(G) \\ e \in E(G)}}$  which is defined by:

$$a_{v,e} = \begin{cases} 1, & \text{if } v \in e \\ 0, & \text{if } v \notin e \end{cases}$$

The **incidence matrix** of a directed graph  $G$  is the matrix  $A_G = (a_{v,e})_{\substack{v \in V(G) \\ e \in E(G)}}$  which is defined by:

$$a_{v,(x,y)} = \begin{cases} -1, & \text{if } v = x \\ 1, & \text{if } v = y \\ 0, & \text{if } v \notin \{x, y\} \end{cases}$$

**Theorem 85** *The incidence matrix of an undirected graph  $G$  is totally unimodular if and only if  $G$  is bipartite.*

**Proof:** Let  $G$  be an undirected graph and  $A_G$  its incidence matrix. Since a matrix is TU if and only if its transposed matrix is TU, we can apply Theorem 84 to the rows of  $A_G$ :  $A_G$  is TU if and only if for each  $X \subseteq V(G)$  there is a partition  $X = A \dot{\cup} B$  with  $E(G[A]) = E(G[B]) = \emptyset$ . The last condition is satisfied if and only if  $G$  is bipartite.  $\square$

### Applications:

- The previous theorem can be used to show **König's Theorem**: The maximum cardinality of a matching in a bipartite graph equals the minimum cardinality of a vertex cover. To see this, let  $G$  be a bipartite graph and  $A_G$  its incident matrix. Then, a maximum matching is given by an integral solution of  $\max\{\mathbb{1}_m x \mid A_G x \leq \mathbb{1}_n, x \geq 0\}$  and a minimum vertex cover by an integral solution of  $\min\{\mathbb{1}_n y \mid A_G^t y \geq \mathbb{1}_m, y \geq 0\}$ . By the previous theorem,  $A_G$  is TU, so by Corollary 82 both optima are attained by integral vectors.
- Another implication of the theorem provides a characterization of **doubly stochastic matrices**: A square matrix  $M = (x_{ij})_{\substack{i=1,\dots,n \\ j=1,\dots,n}} \in \mathbb{R}_{\geq 0}^{n \times n}$  is called doubly stochastic if for all  $i \in \{1, \dots, n\}$ , we have  $\sum_{j=1}^n x_{ij} = 1$  and for all  $j \in \{1, \dots, n\}$ , we have  $\sum_{i=1}^n x_{ij} = 1$ . If in addition all entries are integral, we call the matrix a **permutation matrix**. We claim that each doubly stochastic matrix can be written as a convex combination of permutation matrices (which has also been proved in an earlier exercise). To see this, note that the set of all doubly stochastic  $n \times n$ -matrices is given by  $P = \{x \in \mathbb{R}^{n^2} \mid A_G x \leq \mathbb{1}, x \geq 0\}$  where  $A_G$  is the incidence vector of the complete bipartite Graph  $K_{n,n}$  (which contains a vertex for each column of  $M$  in one side of the bipartition and a vertex for each row of  $M$  in the other side of the bipartition). Since  $A_G$  is TU, by Theorem 81 all vertices of  $P$  are integral, so the represent permutation matrices.

**Theorem 86** *The incidence matrix of a directed graph is totally unimodular.*

**Proof:** Again, we apply Theorem 84 to the transpose of the incidence matrix. For any set  $R \subseteq \{1, \dots, m\}$  we can choose the  $R_1 := R$  and  $R_2 := \emptyset$  satisfying the constraints of Theorem 84.  $\square$

**Remark:** This result gives a reason for the existence of integral optimum solution of flow problems. These results can be extended to more general linear functions on the edges of directed graphs (see exercises).

## 8.5 Cutting Planes

The general strategy of cutting-plane methods can be described as follows: Assume that we are given a polyhedron  $P$  and we want to optimize a linear function over the integral vectors in  $P$ . To this end, we first find an optimum solution  $x^*$  over  $P$ . If this belongs to  $P_I$ , we are done, because then we can also easily compute an integral solution of the same cost. Otherwise we

look for an hyperplane separating  $x^*$  from  $P_I$ , so we ask for a vector  $c$  and a number  $\delta$ , such that  $c^t x \leq \delta$  for all  $x \in P_I$  but  $c^t x^* > \delta$ . Then, we add the constraint  $c^t x \leq \delta$ , solve the linear program again and iterate these steps until we get an integral solution.

How can we find half-space that contain  $P_I$  but not necessarily  $P$ ? An easy observation is that if  $H$  is a half-space that contains  $P$ , then  $P_I$  is contained in  $H_I$ . This motivates the following definition:

**Definition 24** Let  $P \subseteq \mathbb{R}^n$  be a convex set. Let  $M$  be the set of all rational half-spaces  $H = \{x \in \mathbb{R}^n \mid c^t x \leq \delta\}$  with  $P \subseteq H$ . Then, we define

$$P' := \bigcap_{H \in M} H_I.$$

We set  $P^{(0)} := P$  and  $P^{(i+1)} := (P^{(i)})'$  for  $i \in \mathbb{N} \setminus \{0\}$ .  $P^{(i)}$  is the  $i$ -th **Gomory-Chvátal-truncation** of  $P$ .

Obviously, we have  $P \supseteq P^{(1)} \supseteq P^{(2)} \supseteq \dots \supseteq P_I$  for any rational polyhedron  $P$ . In particular we have  $P = P'$  if  $P = P_I$ .

An example that  $P'$  may differ from  $P_I$  is given by the polytope  $P = \text{conv}(\{(0, 0), (0, 1), (1, \frac{1}{2})\})$ . For any half-space  $H$  containing  $P$ , we have  $(\frac{1}{2}, \frac{1}{2}) \in H_I$ , so we get  $(\frac{1}{2}, \frac{1}{2}) \in P'$  and thus  $P_I \neq P'$ . In this polyhedron,  $P_I = P^{(2)}$ , but by extending the polyhedron to the right, one can get for each  $k$  a rational polyhedron for which also  $P_I \neq P^{(k)}$ .

**Lemma 87** Let  $H = \{x \in \mathbb{R}^n \mid c^t x \leq \delta\}$  be a rational half-space with  $c \in \mathbb{Z}^n$  such that the greatest common divisor of the components of  $c$  is 1. Then  $H_I = H' = \{x \in \mathbb{R}^n \mid c^t x \leq \lfloor \delta \rfloor\}$ .

**Proof:** Obviously, we have  $H_I \subseteq H' \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \lfloor \delta \rfloor\}$ , so we only have to show that  $\{x \in \mathbb{R}^n \mid c^t x \leq \lfloor \delta \rfloor\} \subseteq H_I$ . Let  $x^* \in \{x \in \mathbb{R}^n \mid c^t x \leq \lfloor \delta \rfloor\}$ . By Corollary 69, the hyperplane  $\{x \in \mathbb{R}^n \mid c^t x = \lfloor \delta \rfloor\}$  contains an integral vector  $y$  (because the greatest common divisor of the components of the components of  $c$  is 1). Let  $\alpha \in \mathbb{N} \setminus \{0\}$  be a number such that  $\alpha x^*$  is integral. Then,

$$x^* = \frac{1}{\alpha}(\alpha x^* - (\alpha - 1)y) + \frac{\alpha - 1}{\alpha}y.$$

Since  $c^t(\alpha x^* - (\alpha - 1)y) \leq c^t y = \lfloor \delta \rfloor$ , this shows that  $x^*$  is the convex combination of two integral vectors in  $H$ , so  $x^* \in H_I$ .  $\square$

**Proposition 88** *Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  be a rational polyhedron. Then*

$$P' = \{x \in \mathbb{R}^n \mid u^t Ax \leq \lfloor u^t b \rfloor \text{ for all } u \geq 0 \text{ with } u^t A \text{ integral}\}.$$

**Proof:** “ $\subseteq$ :” For any  $u \geq 0$ , we have  $P \subseteq \{x \in \mathbb{R}^n \mid u^t Ax \leq u^t b\}$ . Hence, if in addition  $u^t A$  is integral, this implies  $P' \subseteq \{x \in \mathbb{R}^n \mid u^t Ax \leq u^t b\}_I \subseteq \{x \in \mathbb{R}^n \mid u^t Ax \leq \lfloor u^t b \rfloor\}$ .

“ $\supseteq$ :” W.l.o.g. we can assume that  $\{x \in \mathbb{R}^n \mid u^t Ax \leq \lfloor u^t b \rfloor \text{ for all } u \geq 0 \text{ with } u^t A \text{ integral}\} \neq \emptyset$ . Then also  $P \neq \emptyset$ .

Let  $z \in \{x \in \mathbb{R}^n \mid u^t Ax \leq \lfloor u^t b \rfloor \text{ for all } u \geq 0 \text{ with } u^t A \text{ integral}\}$ . We have to show that  $z$  is in  $P'$ , i.e. that  $z$  is contained in the integer hull of every half-space containing  $P$ .

Let  $H = \{x \in \mathbb{R}^n \mid c^t x \leq \delta\}$  with  $c \in \mathbb{Q}^n$  such that  $P \subseteq H$ . We can assume that the greatest common divisor of the components of  $c$  is 1.

The LP  $\max\{c^t x \mid Ax \leq b\}$  is feasible and bounded (by  $\delta$ ), so we get the duality equation

$$\max\{c^t x \mid Ax \leq b\} = \min\{u^t b \mid A^t u = c, u \geq 0\}.$$

Let  $\tilde{u}$  be an optimum solution of the minimum. Since  $\tilde{u}^t A = c^t$  is integral, this leads to  $\tilde{u}^t A z \leq \lfloor \tilde{u}^t b \rfloor$ , so

$$c^t z = \tilde{u}^t A z \leq \lfloor \tilde{u}^t b \rfloor \leq \delta.$$

By the previous lemma, this implies  $z \in H_I$ . Since this is true for any half-space  $H$  containing  $P$ , it also shows  $z \in P'$ .  $\square$

Cuts that are given by inequalities of the type  $u^t Ax \leq \lfloor u^t b \rfloor$  (for some vector  $u \geq 0$  with  $u^t A$  integral) are called **Gomory-Chvátal cuts**. They have been used for the first algorithms for integer linear programming based on cutting planes (see Gomory [1963]).

**Theorem 89** *Let  $Ax \leq b$  with  $A \in \mathbb{Z}^{m \times n}$  and  $b \in \mathbb{Q}^m$  be a TDI-system. Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ . Then,  $P' = \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\}$ .*

**Proof:** “ $P' \subseteq \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\}$ :” Each inequality in  $Ax \leq b$  gives a half-space  $H$ , and the corresponding inequality in  $Ax \leq \lfloor b \rfloor$  gives a half-space that contains  $H_I$  and hence  $P'$ .

“ $P' \supseteq \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\}$ :” We can assume that  $\{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\} \neq \emptyset$ . Let  $\tilde{x} \in \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\}$ , and let  $u \geq 0$  be a vector with  $u^t A$  integral. By the previous proposition, we have to show that  $u^t A \tilde{x} \leq \lfloor u^t b \rfloor$ .

The LP  $\max\{u^t Ax \mid Ax \leq b\}$  is feasible (since  $\{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\} \neq \emptyset$ ) and bounded (by  $u^t b$ ), so we have the primal-dual equation

$$\max\{u^t Ax \mid Ax \leq b\} = \min\{b^t y \mid y \geq 0, y^t A = u^t A\}.$$

Since  $Ax \leq b$  is TDI, the minimum is attained by an integral vector  $\tilde{y}$ . Thus,

$$u^t A \tilde{x} = \tilde{y}^t A \tilde{x} \leq \tilde{y}^t \lfloor b \rfloor \leq \lfloor \tilde{y}^t b \rfloor \leq \lfloor u^t b \rfloor.$$

This shows  $P' \supseteq \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor\}$ . □

**Corollary 90** *For any rational polyhedron  $P$ , the set  $P'$  is a polyhedron.*

**Proof:** Follows from the previous theorem and the fact that any rational polyhedron can be described by a TDI-system with integral matrix (Theorem 77). □

**Lemma 91** *Let  $F$  be a face of a rational polyhedron  $P$ . Then,  $F' = F \cap P'$ .*

**Proof:** Let  $P$  be a rational polyhedron. By Theorem 77, we can write  $P$  as  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A$  integral,  $b$  rational and  $Ax \leq b$  TDI. Let  $F = \{x \in \mathbb{R}^n \mid Ax \leq b, a^t x = \beta\}$  be a face of  $P$  where  $a^t x \leq \beta$  with  $a$  and  $\beta$  integral is a valid inequality for  $P$ . By Proposition 72, the system  $Ax \leq b, a^t x \leq \beta$  is TDI. Therefore, by Proposition 73, also  $Ax \leq b, a^t x = \beta$  is TDI. Since  $\beta$  is integral, we get (by applying Theorem 89 twice):

$$\begin{aligned} P' \cap F &= \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor, a^t x = \beta\} \\ &= \{x \in \mathbb{R}^n \mid Ax \leq \lfloor b \rfloor, a^t x \leq \lfloor \beta \rfloor, a^t x \geq \lceil \beta \rceil\} \\ &= F'. \end{aligned}$$

□

**Corollary 92** *Let  $F$  be a face of a rational polyhedron  $P$ . Then,  $F^{(i)} = F \cap P^{(i)}$ .*

**Proof:** Let  $P$  be a rational polyhedron, and  $F$  a face of  $P$ . By the previous lemma,  $F'$  is either empty or a face of  $P'$ . By induction on  $i$ , we show that  $F^{(i)}$  is either empty or a face of  $P^{(i)}$ , and  $F^{(i)} = F \cap P^{(i)}$ . For  $i = 1$ , this follows from that previous lemma. For  $i > 1$  we get:  $F^{(i)} = (F^{(i-1)})' = (P^{(i-1)})' \cap F^{(i-1)} = P^{(i)} \cap (P^{(i-1)} \cap F) = P^{(i)} \cap F$ . □

**Lemma 93** *Let  $P \subseteq \mathbb{R}^n$  be a polyhedron,  $U$  a unimodular  $n \times n$ -matrix and  $f(X) = \{Ux \mid x \in X\}$  for all  $X \subseteq \mathbb{R}^n$ . Then,  $f(P)$  is a polyhedron. Moreover, if  $P$  is rational, then  $(f(P))' = f(P')$  and  $(f(P))_I = f(P_I)$ .*

**Proof:** Let  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ , then  $f(P) = \{x \in \mathbb{R}^n \mid AU^{-1}x \leq b\}$ , so  $f(P)$  is a polyhedron.

Now assume in addition that  $P$  is rational. Since  $U$  is unimodular,  $Ux$  is integral if and only if  $x$  is integral. This implies

$$\begin{aligned}
(f(P))_I &= \text{conv}(\{y \in \mathbb{Z}^n \mid y = Ux, x \in P\}) \\
&= \text{conv}(\{y \in \mathbb{R}^n \mid y = Ux, x \in P, x \in \mathbb{Z}^n\}) \\
&= \text{conv}(\{y \in \mathbb{R}^n \mid y = Ux, x \in P_I\}) \\
&= f(P_I).
\end{aligned}$$

By Theorem 77, we can assume that  $Ax \leq b$  is TDI,  $A$  is integral and  $b$  is rational. Then, for any integral vector  $c$  for which  $\min\{b^t y \mid y^t A U^{-1} = c^t, y \geq 0\}$  is feasible and bounded, also  $\min\{b^t y \mid y^t A = c^t U, y \geq 0\}$  is feasible and bounded and  $c^t U$  is integral. Hence  $AU^{-1}x \leq b$  is TDI. Thus, Theorem 89 implies

$$(f(P))' = \{x \in \mathbb{R}^n \mid AU^{-1}x \leq b\}' = \{x \in \mathbb{R}^n \mid AU^{-1}x \leq \lfloor b \rfloor\} = f(P').$$

□

**Remark:** This shows as well that  $(f(P))^{(i)} = f(P^{(i)})$  for a rational polyhedron  $P$  and  $i \in \mathbb{N}$ .

**Theorem 94** *For every rational polyhedron  $P$ , there is a number  $t$  with  $P^{(t)} = P_I$ .*

**Proof:** Let  $P \subseteq \mathbb{R}^n$  be a rational polyhedron. We prove the statement by induction on  $n + \dim(P)$ . The case  $\dim(P) = 0$  is trivial.

Case 1:  $\dim(P) < n$ .

Then,  $P \subseteq K$  for some rational hyperplane  $K = \{x \in \mathbb{R}^n \mid a^t x = \beta\}$ . We can assume that the greatest common divisor of the entries of  $a$  is 1.

If  $K$  does not contain any integral vector, then by Corollary 69,  $\beta$  must be non-integral. Then,  $P' \subseteq \{x \in \mathbb{R}^n \mid a^t x \leq \lfloor \beta \rfloor, a^t x \geq \lceil \beta \rceil\} = \emptyset = P_I$ .

If  $K$  contains an integral vector  $y$ , we can assume that it contains 0 because the theorem holds for  $P$  if and only if it holds for  $P - y$  since  $y$  is integral. Thus, we can assume that  $\beta = 0$ .

If we interpret  $a^t$  as a  $1 \times n$ -matrix, we can bring it into Hermite normal form by elementary unimodular column operations. The Hermite normal form of  $a^t$  is of the type  $\alpha e_1^t$ . Since any series of elementary unimodular column operations can be performed by a multiplication from the right with a unimodular square matrix, there is a unimodular square matrix  $U$  with  $a^t U = \alpha e_1^t$ . However, by the previous lemma, the theorem is invariant under the transformation  $x \mapsto U^{-1}x$ , so we may assume that  $a^t = \alpha e_1^t$ . Then, the first component of every vector in  $P$  is zero, so  $P = \{0\} \times Q$  for some polyhedron  $Q \subseteq \mathbb{R}^{n-1}$ . We can apply the induction hypothesis to  $Q$ . Since  $(\{0\} \times Q)_I = \{0\} \times Q_I$  and  $(\{0\} \times Q)^{(t)} = \{0\} \times Q^{(t)}$  for any  $t \in \mathbb{N}$ , this proves the theorem in the case  $\dim(P) < n$ .

Case 2:  $\dim(P) = n$ . We can write  $P$  as  $P = \{x \in \mathbb{R}^n \mid Ax \leq b\}$  with  $A$  integral. Since  $P$  is rational, by Theorem 66,  $P_I$  is a rational polyhedron as well, so it can be written as

$P_I = \{x \in \mathbb{R}^n \mid Cx \leq d\}$  with some integral matrix  $C$  and some rational vector  $d$ . If  $P_I = \emptyset$ , we choose  $C = A$  and  $d = b - A'\mathbb{1}_n$  where  $A'$  arises from  $A$  by taking the absolute value of each entry. Note that  $\{x \in \mathbb{R}^n \mid Ax + A'\mathbb{1}_n \leq b\} = \emptyset$  because any vector  $x^*$  with  $Ax^* + A'\mathbb{1}_n \leq b$  could be rounded down to an integral vector  $x$  with  $Ax \leq b$ .

Let  $c^t x \leq \delta$  be an inequality in  $Cx \leq d$ . Then, we claim that there is an  $s \in \mathbb{N}$  with  $P^{(s)} \subseteq H := \{x \in \mathbb{R}^n \mid c^t x \leq \delta\}$ . The theorem is a direct consequence of this claim.

Proof of the claim: Observe that there is a number  $\beta \geq \delta$  with  $P \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \beta\}$ . If  $P_I = \emptyset$ , this is true by construction. In the case  $P_I \neq \emptyset$ , it follows from the fact that  $c^t x$  is bounded over  $P$  if and only if it is bounded over  $P_I$  (Proposition 67).

Assume that the claim is false, so there is an integer  $\gamma$  with  $\delta < \gamma \leq \beta$  for which there is an  $s_0 \in \mathbb{N}$  with  $P^{(s_0)} \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \gamma\}$  but there is no  $s \in \mathbb{N}$  with  $P^{(s)} \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \gamma - 1\}$ . Then,  $\max\{c^t x \mid x \in P^{(s)}\} = \gamma$  for all  $s \geq s_0$ . To see this, assume that  $\max\{c^t x \mid x \in P^{(s)}\} < \gamma$  for some  $s$ . Then there is an  $\epsilon > 0$  with  $P^{(s)} \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \gamma - \epsilon\}$ . This implies  $\max\{c^t x \mid x \in P^{(s+1)}\} \leq \gamma - 1$  because  $\{x \in \mathbb{R}^n \mid c^t x \leq \gamma - \epsilon\}_I \subseteq \{x \in \mathbb{R}^n \mid c^t x \leq \gamma - 1\}$ .

Define  $F := P^{(s_0)} \cap \{x \in \mathbb{R}^n \mid c^t x = \gamma\}$ . Then,  $\dim(F) < n = \dim(P)$ , so we can apply the induction hypothesis to  $F$ , which implies that there is a number  $s_1$  with  $F^{(s_1)} = F_I$ . Thus,

$$F^{(s_1)} = F_I \subseteq P_I \cap \{x \in \mathbb{R}^n \mid c^t x = \gamma\} = \emptyset.$$

Since  $F$  is a face of  $P^{(s_0)}$ , we can apply Corollary 92 to  $F$  and  $P^{(s_0)}$ , so

$$\emptyset = F^{(s_1)} = P^{(s_0+s_1)} \cap F = P^{(s_0+s_1)} \cap \{x \in \mathbb{R}^n \mid c^t x = \gamma\}.$$

Therefore,  $\max\{c^t x \mid x \in P^{(s_0+s_1)}\} < \gamma$ , which is a contradiction.  $\square$

**Theorem 95** For every polytope  $P$ , there is a number  $t$  with  $P^{(t)} = P_I$ .

**Proof:** We claim that there is a rational polytope  $Q$  with  $P \subseteq Q$  such that  $Q_I = P_I$ . This is sufficient to show the theorem because we can apply the previous theorem to  $Q$ . To prove the claim, let  $W$  be a hypercube containing  $P$ . Then, there are finitely many integral vectors  $z \in W \setminus P$ . For each such vector  $z$  we choose a rational hyperplane separating  $z$  from  $P$ . The set of all the inequalities corresponding to these separating hyperplanes together with the inequalities that define  $W$  give a description of a set  $Q$  with the desired properties.  $\square$

For a polyhedron  $P$ , the smallest number  $t$  with  $P^{(t)} = P_I$  (if there is such a number) is called **Chvátal rank** of  $P$ .

## 8.6 Branch-and-Bound Methods

Note that this section was not covered by the lecture course given in summer term 2019.



BRANCH-AND-BOUND METHODS (they are also called DIVIDE-AND-CONQUER ALGORITHMS or BACKTRACKING ALGORITHMS) are a quite simple approach to integer linear programming. Nevertheless, they are of great practical relevance. Algorithm 4 describes the approach for integer linear programs but it can be applied to mixed integer linear programs, too. The algorithm stores a number  $L$  which is the cost of the best integral solution found so far (so in the beginning it is  $-\infty$ ). In each iteration of the main loop, the algorithm chooses a polyhedron  $P_j$ , which is a subset of the given polyhedron  $P_0$ , and solves the corresponding linear program. If this LP is bounded and feasible, the algorithm first checks if the value  $c^*$  of an optimum solution  $x^*$  is larger than  $L$ . If this is not the case, the algorithm can reject the polyhedron  $P_j$  because it cannot contain a better integral solution than the best current solution (this is the *bounding* part). If  $c^* > L$  and  $x^*$  is integral, we have found a better integral solution and can update  $L$ . Otherwise, we choose a non-integral component  $x_i^*$  of  $x^*$  and compute sub-polyhedra  $P_{2j+1}$  and  $P_{2j+2}$  of  $P_j$  with additional constraints that arise by rounding  $x_i^*$  up or down (*branching* step).

---

**Algorithm 4:** Branch-and-Bound Algorithm

---

**Input:** A matrix  $A \in \mathbb{Q}^{m \times n}$ , a vector  $b \in \mathbb{Q}^m$ , and a vector  $c \in \mathbb{Q}^n$  such that the LP  $\max\{c^t x \mid Ax \leq b\}$  is feasible and bounded.

**Output:** A vector  $\tilde{x} \in \{x \in \mathbb{Z}^n \mid Ax \leq b\}$  maximizing  $c^t x$  or the message that there is no optimum solution.

```

1   $L := -\infty$ ;
2   $P_0 := \{x \in \mathbb{R}^n \mid Ax \leq b\}$ ;
3   $\mathcal{K} := \{P_0\}$ ;
4  while  $\mathcal{K} \neq \emptyset$  do
5      Choose a  $P_j \in \mathcal{K}$ ;
6       $\mathcal{K} := \mathcal{K} \setminus \{P_j\}$ ;
7      if  $P_j \neq \emptyset$  then
8          Solve  $\max\{c^t x \mid x \in P_j\}$ ;
9          Let  $x^*$  be an optimum solution and  $c^* = c^t x^*$ ;
10         if  $c^* > L$  then
11             if  $x^* \in \mathbb{Z}^n$  then
12                  $L := c^*$ ;
13                  $\tilde{x} := x^*$ ;
14             else
15                 Choose  $i \in \{1, \dots, n\}$  with  $x_i^* \notin \mathbb{Z}$ ;
16                  $P_{2j+1} := \{x \in P_j \mid x_i \leq \lfloor x_i^* \rfloor\}$ ;
17                  $P_{2j+2} := \{x \in P_j \mid x_i \geq \lceil x_i^* \rceil\}$ ;
18                  $\mathcal{K} := \mathcal{K} \cup \{P_{2j+1}\} \cup \{P_{2j+2}\}$ ;
19 if  $L > -\infty$  then
20     return  $\tilde{x}$ ;
21 else
22     return "There is no feasible solution";

```

---

**Example:** Consider the following ILP:

$$\begin{aligned}
 \max \quad & -x_1 + 3x_2 \\
 \text{subject to} \quad & -4x_1 + 6x_2 \leq 9 \\
 & x_1 + x_2 \leq 4 \\
 & x_1, x_2 \geq 0 \\
 & x_1, x_2 \in \mathbb{Z}
 \end{aligned}$$

Figure 9 illustrates what the algorithm may do on this instance. Since the optimum solution of the LP-relaxation is not integral, we create in the first branching step two sub-polytopes  $P_1 = \{(x_1, x_2) \mid x_2 \leq 2\} \cap P_0$  and  $P_2 = \{(x_1, x_2) \mid x_2 \geq 3\} \cap P_0 = \emptyset$ . In  $P_1$  we still do not find an integral optimum solution, so we branch again and get the polytopes  $P_3$  and  $P_4$ . In  $P_4$  we get an integral optimum  $x^* = (1, 2)$  with cost 3. In  $P_3$  we get a non-integral optimum solution  $(0, 1.5)$  whose cost is not better than the best integral solution found so far (provided that we considered  $P_4$  before  $P_3$ ), so the algorithm will stop here.

A branch-and-bound computation is often represented by a so-called **branch-and-bound tree**. This is in fact rather an arborescence than a tree. Its nodes are the polyhedra  $P_j$  that are considered during the computation, and  $P_0$  is the root. For any  $P_j$ , the nodes  $P_{2j+1}$  and  $P_{2j+2}$  are its children (if they exist).

In line 5 of the algorithm, we have to choose the next LP to be solved, and in line 15 we have to decide which non-integral component is used for creating new sub-problems. There are different strategies for these steps (branching rules). For example, it is often reasonable to store the elements of  $\mathcal{K}$  in a last-in-first-out queue and to choose the last element that has been added to  $\mathcal{K}$ . In the branch-and-bound tree, this corresponds to a leaf with the biggest distance to the root. This strategy can reduce the time until the first feasible solution has been found. Another reasonable branching rule consist in choosing a polyhedron  $P_j$  for which  $\max\{c^t x \mid x \in P_j\}$  is as large as possible. Note that the maximum over all these values for all  $P_j \in \mathcal{K}$  gives an upper bound  $U$  on the best possible solution that can still be computed. Hence, by choosing a  $P_j$  with  $\max\{c^t x \mid x \in P_j\} = U$ , we get a chance to reduce  $U$ . This can be useful if we do not want to compute an exact optimum solution but we stop as soon as  $U - L$  is small enough.

For the choice of  $x_i^*$  a common strategy is to choose  $x_i^*$  such that  $|x_i^* - \lfloor x_i^* \rfloor - \frac{1}{2}|$  is minimized. Another, more time-consuming approach is to choose  $x_i^*$  such that the effect on the objective function is maximized (strong branching).

**Further remarks:**

- In order to get at least a finite algorithm, we have to guarantee that in line 8 we always find a integral optimum solution if  $P_j$  is integral.
- Instead of initializing  $L$  with  $-\infty$ , it is often possible to compute some reasonable integral solution by some heuristics. In particular this is often the case for combinatorial problems.
- The branch-and-bound strategy can be combined with a cutting-plane algorithm (see the previous section). For each sub-polyhedron  $P_j$ , one can try to find hyperplanes separating

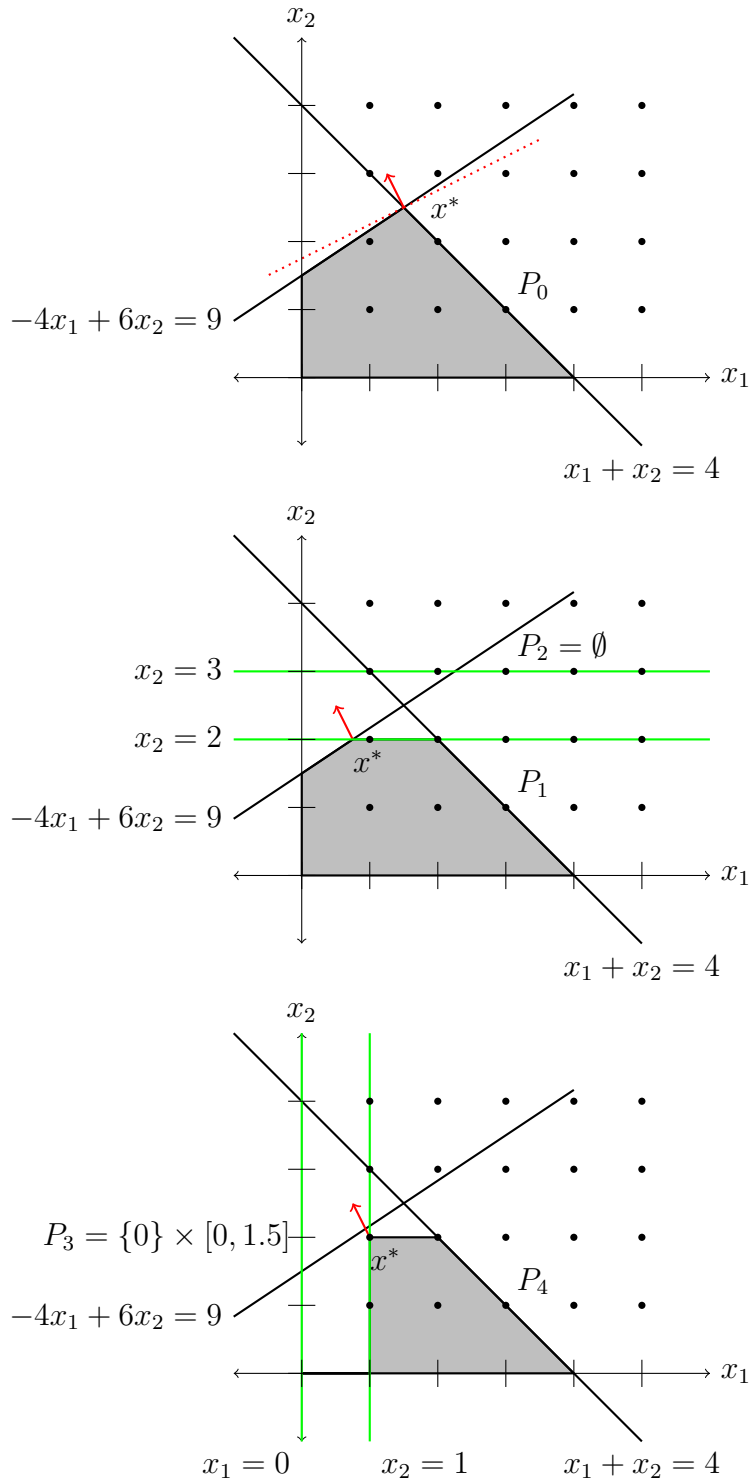


Fig. 9: A branch-and-bound example.

some non-integral vectors in  $P_j$  from  $(P_j)_I$ . This combination is called **branch-and-cut method**. For example, this approach has been for solving quite large Traveling Salesman Problems (see Padberg and Rinaldi [1991]).

## Bibliography

- Adler, I., Karp, R.M., Shamir, R. [1987]: *A simplex variant solving an  $m \times d$  linear program in  $O(\min(m^2, d^2))$  expected number of steps*. Journal of Complexity, 3, 372–387, 1987.
- Ahuja, R.K., Magnanti, T.L., and Orlin [1993]: *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, 1993.
- Anthony, M., and Harvey, M. [2012]: *Linear Algebra: Concepts and Methods*. Cambridge University Press, 2012.
- Bárány, I., Howe, R., and Lovász, L. [1992]: *On integer points in polyhedra: a lower bound*. Combinatorica, 12, 135–142, 1992.
- Bertsimas, D., and Tsitsiklis, J.N. [1997]: *Introduction to Linear Optimization*. Athena Scientific, 1997.
- Bertsimas, D., and Weismantel, R. [2005]: *Optimization over Integers*. Dynamic Ideas, 2005.
- Bland, R.G. [1977]: *New finite pivoting rules for the simplex method*. Mathematics of Operations Research, 2, 103–107, 1977.
- Borgwardt, K. [1982]: *The average number of pivot steps required by the simplex method is polynomial*. Zeitschrift für Operations Research, 26, 157–177, 1982.
- Bosch, S. [2007]: *Lineare Algebra*. 4th edition, Springer, 2007.
- Chvátal, V. [1983]: *Linear programming*. Series of books in the mathematical sciences, W.H. Freeman, 1983.
- Cohn, D.L. [1980]: *Measure Theory*. Birkhäuser, 1980.
- Cunningham, W.H. [1976]: *A network simplex method*. Mathematical Programming, 11, 105–116, 1976.
- Dadush, D. and Hulberts, S. [2019]: *A Friendly Smoothed Analysis of the Simplex Method*. SIAM Journal on Computing, 49, 5, 449–499, 2019
- Dantzig, G.B. [1951]: *Maximization of a linear function of variables subject to linear inequalities*. In: Koopmans, T.C (ed.), Activity Analysis of Production and Allocation, 359–373, Wiley, 1951.
- Edmonds, J. [1965]: *Maximum matching and polyhedron with  $(0,1)$  vertices*. Journal of Research of the National Bureau of Standards, B, 69, 125–130, 1965.
- Eisenbrand, F. [2003]: *Fast integer programming in fixed dimension*. Lecture Notes in Computer Science, 2832, 196–207, 2003.

- Fischer, G. [2009]: *Lineare Algebra: Eine Einführung für Studienanfänger*. 18th edition, Springer, 2013.
- Ghoulia-Houri, A. [1962]: *Caractérisation des matrices totalement unimodulaires*. Comptes Rendus Hebdomadaires des Séances de l'Académie des Sciences (Paris), 254, 1192-1194, 1962.
- Giles, F.R. and Pulleyblank, W.R. [1979]: *Total dual integrality and integer polyhedra*. Linear Algebra and Its Applications, 25, 191–196, 1979.
- Gomory, R.E. [1963]: *An algorithm for integer solutions of linear programs*. In: Recent Advances in Mathematical Programming (R.L. Graves, P. Wolfe, eds.), McGraw-Hill, 269–302, 1963.
- Grötschel, M., Lovász, L. and Schrijver, A. [1981]: *The ellipsoid method and its consequences in combinatorial optimization*. Combinatorica, 1, 169–197, 1981.
- Guenin, B., Könemann, J., and Tunçel, L. [2014]: *A Gentle Introduction to Optimization*. Cambridge University Press, 2014.
- Hoffman, A. and Kruskal, J. [1956]: *Integral boundary points of convex polyhedra*. Linear Inequalities and Related Systems (H. Kuhn, A. Tucker, eds.), Annals of Mathematics Studies, 38, 223–246, 1956.
- Hougardy, S., and Vygen, J. [2018]: *Algorithmische Mathematik*. Second edition, Springer, 2018.
- Kalai, G., and Kleitman, D. [1992]: *A quasi-polynomial bound for the diameter of graphs of polyhedra*. Bulletin of the American Mathematical Society, 26, 315–316, 1992.
- Karmakar, L. [1984]: *A new polynomial-time algorithm for linear programming*. Combinatorica, 4, 373–395, 1984.
- Karloff, H. [1991]: *Linear Programming*. Birkhäuser, 1991.
- Khachiyan, L. [1979]: *A polynomial algorithm for linear programming*. Soviet Mathematics Doklady, 20, 191–194, 1979.
- Klee, V., and Minty, G.J. [1972]: *How good is the simplex algorithm?* In: Inequalities III (O. Shisha, ed.), Academic Press, 159–175, 1972.
- Korte, B., and Vygen, J. [2018]: *Combinatorial Optimization: Theory and Algorithms*. Sixth edition, Springer, 2018.
- Lang, S. [1987]: *Linear Algebra*. Third edition, Springer, 1987.
- Lee, T., Sidford, A., Wong, S.C. [2015]: *A Faster Cutting Plane Method and its Implications for Combinatorial and Convex Optimization*. arxiv.org/abs/1508.04874, Symposium on Foundations of Computer Science, 2015.
- Lenstra, H.W. [1983]: *Integer programming with a fixed number of variables*. Mathematics of Operations Research, 8, 538–548, 1983.

- Matoušek, J., and Gärtner, B. [2007]: *Understanding and Using Linear Programming*. Springer, 2007.
- Megiddo, N. [1984]: *Linear programming in linear time when the dimension is fixed*. Journal of the ACM, 31, 114–127, 1984.
- Mehlhorn, K., and Saxena, S. [2015]: *A still simpler way of introducing the interior-point method for linear programming*. Computer Science Review, 22, 1–11, 2016.
- Padberg, M. [1999]: *Linear Optimization and Extensions*. Second edition, Springer, 1999
- Padberg, M., and Rao, M. [1982]: *Odd minimum cut-sets and b-matchings*. Mathematics of Operations Research, 7, 67–80, 1982.
- Padberg, M., and Rinaldi, G. [1991]: *A Branch-and-Cut Algorithm for the Resolution of Large-Scale Symmetric Traveling Salesman Problems*. SIAM Review, 33, 1, 60–100, 1991.
- Pan, P.-Q. [2014]: *Linear Programming Computation*. Springer, 2014.
- Panik, M.J. [1996]: *Linear Programming: Mathematics, Theory and Algorithms*. Kluwer Academic Publishers, 1996.
- Roos, C., Terlaky, T., Vial, J.-P. [2005]: *Interior Point Methods for Linear Optimization*. Second edition, Springer, 2005.
- Rubin, D. [1970]: *On the unlimited number of faces in integer hulls of linear programs with a single constraint*. Operations Research, 18, 5, 940 – 946, 1970.
- Saigal, R. [1995]: *Linear Programming. A Modern Integrated Analysis*. Springer, 1995.
- Santos, F. [2011]: *A counterexample to the Hirsch conjecture*. Annals of Mathematics, 176, 1, 383–412, 2011.
- Schrijver, A. [1986]: *Theory of Linear and Integer Programming*. Wiley, 1986.
- Sierksma, G., and Zwols, Y. [2015]: *Linear and Integer Optimization. Theory and Practice*. Third edition, CRC Press, 2015.
- Spielmann, D.A., and Teng, S.-H. [2005]: *Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time*. Journal of the ACM, 51, 3, 385 – 463, 2004.
- Strang, G. [1980]: *Linear Algebra and Its Applications*. Second edition, Academic Press, 1980.
- Tardos, É. [1986]: *A strongly polynomial algorithm to solve combinatorial linear programs*. Operations Research, 34, 2, 250 – 256, 1986.
- Terlaky, R.J. [2001]: *An easy way to teach interior point methods*. European Journal of Operational Research, 130, 1–19, 2001.

- Vanderbei, R.J. [2014]: *Linear Programming: Foundations and Extensions*. Fourth edition, Springer, 2014.
- Wright, S.J. [1997]: *Primal-Dual Interior-Point Methods*. SIAM, 1997.
- Ye, Y. [1992]: *On the finite convergence of interior-point algorithms for linear programming*. *Mathematical Programming*, 57, 325–335, 1992.
- Ye, Y. [1997]: *Interior Point Algorithms. Theory and Analysis*. Wiley, 1997.
- Ziegler, G. [2007]: *Lectures on Polytopes*. Seventh Printing, Springer, 2007.



# Index

- Active row, 100
- Affine linear mapping, 31
  
- Barrier function, 91
- Barrier problem, 91
- Basic solution, 32
- Basic variables, 44
- Basis, 44
- BINARY LINEAR PROGRAMS, 93
- Bland's rule, 53
- Bounded optimization problem, 4
- BRANCH-AND-BOUND ALGORITHM, 113
- Branch-and-bound tree, 114
- Branch-and-cut method, 116
- Branching rule, 114
  
- Carathéodory's Theorem, 37
- Central path, 92
- Cholesky decomposition, 63
- Chvátal rank, 112
- Column generation, 56
- Combinatorial diameter, 55
- Complementary slackness, 25
- Convex, 10
- Convex combination, 11
- Convex cone, 12
- Convex hull, 11
- Cycling, 52
  
- Degenerated feasible basic solution, 44
- Dimension, 12
- Doubly stochastic matrix, 107
- Dual feasible basis, 56
- Dual LP, 15
- DUAL SIMPLEX ALGORITHM, 57
  
- Elementary unimodular column operation, 95
- Ellipsoid, 63
- ELLIPSOID ALGORITHM, 71
  
- Face, 32
- Facet, 34
  
- Facet-defining, 34
- Farkas' Lemma, 19
- Farkas-Minkowski-Weyl Theorem, 38
- Feasible basic solution, 44
- Feasible basis, 44
- Finitely generated cone, 13
- Fourier-Motzkin elimination, 18
- Fundamental Theorem of Linear Inequalities, 38
  
- Gaussian Elimination, 43, 61
- Gomory-Chvátal cut, 109
- Gomory-Chvátal-truncation, 108
  
- Half-Ball Lemma, 65
- Half-Ellipsoid Lemma, 66
- Half-space, 11
- Hermite normal form, 95
- Hilbert basis, 99
- Hyperplane, 11
  
- IDEALIZED ELLIPSOID ALGORITHM, 67
- Incidence matrix, 106
- Integer hull, 93
- INTEGER LINEAR PROGRAM, 7
- Integral polyhedron, 95
- Integrality Gap, 10
  
- Klee-Minty cube, 55
- König's Theorem, 107
  
- Lagrange function, 91
- Lagrangian relaxation, 91
- Largest coefficient rule, 52
- Largest increase rule, 52
- LINEAR PROGRAM, 5
- Löwner-John Ellipsoid, 67
- LP-relaxation, 9
  
- Matching, 74
- Max-Flow-Min-Cut-Theorem, 29
- Maximization problem, 4

MAXIMUM-FLOW PROBLEM, 7  
 Minimal face, 35  
 Minimally TDI, 99  
 Minimization problem, 4  
 Minkowski sum, 40  
 MIXED INTEGER LINEAR PROGRAM, 7  
  
 Non-basic variables, 44  
 Non-degenerated feasible basic solution, 44  
 Normal, 11  
  
 Objective function, 4  
 Optimization problem, 4  
  
 Permutation matrix, 107  
 Pivot rule, 52  
 Pointed polyhedron, 37  
 Polar cone, 39  
 Polyhedral cone, 12  
 Polyhedron, 11  
 Polytope, 11  
 Positive definite matrix, 63  
 Positive semidefinite matrix, 63  
 Primal LP, 16  
 Projection of a polyhedron, 31  
  
*r-R*-sandwiched set, 75  
 Reduced cost, 50  
 REVISED SIMPLEX ALGORITHM, 56  
  
*s-t*-cut, 29  
*s-t*-flow, 7  
 Separation oracle, 67  
 SIMPLEX ALGORITHM, 51  
 Simplex tableau, 49  
 Slack variable, 5  
 STABLE SET PROBLEM, 10  
 Standard equation form, 5  
 Standard inequality form, 5  
 Steepest edge rule, 52  
 Strict complementary slackness, 27  
 Supporting hyperplane, 32  
  
 TDI-system, 98  
 Totally dual integral, 98  
 Totally unimodular matrix, 103  
  
 Unbounded optimization problem, 4  
 Unimodular matrix, 103  
  
 Vertex, 32  
 VERTEX COVER PROBLEM, 9  
  
 Weak duality, 16  
 Weak optimization problem, 75  
 Weak separation oracle, 76