IOE 610 class Note

Xuyuan Zhang

Update on May 10, 2024

Contents

List of Theorems

List of Definitions

Part I

Linear Optimization

Chapter 1

Introduction to Linear Programming

1.1 Variants of the Linear Programming Problem

We are given a cost vector $\mathbf{c} = (\mathbf{c_1}, \ldots, \mathbf{c_n})$ and we seek to minimize a linear cost function $\mathbf{c'x} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{c_i x_i}$ over all n-dimensional vectors $\mathbf{x} = (\mathbf{x_1}, \ldots, \mathbf{x_n})$ that satisfy a set of linear constraints. In particular, let M_1, M_2, M_3 be some finite index sets, and suppose that for every i in any one of these sets, we are given an n-dimensional vector $\mathbf{a_i}$ and a scalar b_i, that will be used to form the i-th constraint. Let also N_1 and N_2 be subsets of $\{1, \ldots, n\}$ that indicate which variable x_j are constrainted to be nonnegative or nonpositive, respectively. We can consider the problem:

$$\begin{aligned} \text{minimize} & \mathbf{c'x} \\ \text{subject to} & \mathbf{a_i'x} \geq b_i, \quad i \in M_1 \\ & \mathbf{a_i'x} \leq b_i, \quad i \in M_2 \\ & \mathbf{a_i'x} = b_i, \quad i \in M_3 \\ & x_j \geq 0, \quad j \in N_1 \\ & x_j \leq 0, \quad j \in N_2 \end{aligned}$$

The variables $x_1, \ldots, x_n$ are called *decision variables* and a vector $\mathbf{x}$ satisfying all the constraints is called a *feasible solution* or *feasible vector*. The set of all feasible solutions is called the *feasible set* or *feasible region*. If j is in neither N_1 nor N_2, there are no restrictions on the sign of x_j, in which we say that x_j is free or unrestricted variable. The function $\mathbf{c'x}$ is called the *objective function* or *cost function*. A feasible solution is also called the optimal feasible solution.

Definition 1.1 (standard form)**.** A linear programming problem of the form:

$$\begin{aligned} \text{minimize} & \mathbf{c'x} \\ \text{subject to} & \mathbf{Ax} = \mathbf{b} \\ & \mathbf{x} \geq 0 \end{aligned}$$

Suppose that $\mathbf{x}$ has dimension n and let $\mathbf{A_1}, \ldots, \mathbf{A_n}$ be the columns of $\mathbf{A}$. Then the constraint $\mathbf{Ax} = \mathbf{b}$ is equivalent to the system of linear equations:

$$\sum_{i=1}^{n} \mathbf{A_i x_i} = \mathbf{b}$$

- Given an unrestricted variable x_j in a problem in the general form, we replace it by $x_j^+ - x_j^-$ where x_j^+ and x_j^- are new variables on which we impose from the sign constraints $x_j^+ \geq 0, x_j^- \geq 0$. The underlying idea is that any real number can be written as the difference of two nonnegative numbers.

- Elimination of inequality constraints: Given an inequality constraint of the form:

$$\sum_{j=1}^{n} a_{ij}x_j \leq b_i$$

we introduce a new variable s_i and the standard form constraints $\sum_{j=1}^{n} a_{ij}x_j + s_i = b_i, s_i \geq 0$. Such a variable s_i is called a slake variable. Similarly, an inequality constraint $\sum_{j=1}^{n} a_{ij}x_j \geq b_i$ can be put in standard form by introducing a surplus variable s_i and the constraints $\sum_{j=1}^{n} a_{ij}x_j - s_i = b_i, s_i \geq 0$.

We conclude that a general problem can be brought into standard form and, therefore, we only need to develop methods that are capable of solving standard form problems.

Example 1.1 (Multiperiod planning of electric power capacity)**.** A state wants to plan its elasticity capacity for the next T years, and the state has a forecast of d_t megawatts, presumed accurate, of the demand for elasticity during year $t = 1, \ldots, T$. The existing capacity, which is in oil-fired plants, that will not be retired and will be available during year t, is e_t. There are two alternatives for expanding electric capacity: coal fired or nuclear power plants. There is a capital cost of c_t per megawatt of coal-fired capacity that becomes operational at the beginning of year t. The corresponding capital cost for nuclear power plants is n_t. For various political and safety reasons, it has been decided that no more than 20% of the total capacity should ever be nuclear. Coal plants last for 20 years, while nuclear plants last for 15 years. A least cost capacity expansion plan is desired.

Let x_t and y_t be the amount of coal capacity brought on line at the beginning of year t. Let w_t and z_t be the total coal capacity available in year t. The cost of a capacity expansion plan is:

$$\sum_{t=1}^{T} c_t x_t + n_t y_t$$

Since coal-fired plants last for 20 years, we have:

$$w_t = \sum_{s=\max\{1,\ldots,t-19\}}^{t} x_s, \quad t = 1, \ldots, T$$

Similarly, for nuclear power plants,

$$z_t = \sum_{s=\max\{1,\ldots,t-14\}}^{t} y_s, \quad t = 1, \ldots, T$$

Since the available capacity must meet the forecast demand, we require:

$$w_t + z_t + e_t \geq d_t, \quad t = 1, \ldots, T$$

Finally, since no more than 20% of the total capacity should ever be nuclear, we have:

$$\frac{z_t}{w_t + z_t + e_t} \leq 0.2$$

which can be written as:

$$0.8z_t - 0.2w_t \leq 0.2e_t.$$

Summarizing, the capacity expansion problem is as follows:

$$
\begin{aligned}
\text{minimize} \sum_{t=1}^{T} c_t x_t + n_t y_t & \\
\text{subject to} w_t = & \sum_{s=\max\{1,\ldots,t-19\}}^{t} x_s, \quad t = 1, \ldots, T \\
z_t = & \sum_{s=\max\{1,\ldots,t-14\}}^{t} y_s, \quad t = 1, \ldots, T \\
w_t + z_t + e_t \geq & \, d_t, \quad t = 1, \ldots, T \\
0.8 z_t - 0.2 w_t \leq & \, 0.2 e_t, \quad t = 1, \ldots, T \\
x_t, y_t, w_t, z_t \geq & \, 0, \quad t = 1, \ldots, T
\end{aligned}
$$

Example 1.2 (multicommodity flow problem)**.** Consider a communication network consisting of n nodes. Nodes are connected by communication links. A link allowing one-way transmission from node i to node j is described by an ordered pair (i, j). Let $\mathcal{A}$ be the set of all links. We assume that each link $(i, j) \in \mathcal{A}$ can carry up to u_{ij} bits per second. There is a positive charge c_{ij} per bit transimitted along the link. Each node k generates data, at the rate of $b^{k\ell}$ bits per second, that have to be transmitted to node ℓ, either through a direct link (k, ℓ) or by tracing a sequence of links. The problem is to choose paths along which all data reach their intended destinations, while minizing the total cost.

We introduce variables $x_{ij}^{k\ell}$ indicating the amount of data with origin k and destination ℓ that traverse link (i, j). Let:

$$
b_i^{k\ell} = \begin{cases} b^{k\ell} & \text{if } i = k \\ -b^{k\ell} & \text{if } i = \ell \\ 0 & \text{otherwise} \end{cases}
$$

Thus, $b_i^{k\ell}$ is the net flow at node i, from outside the network, of data with origin k and destination ℓ. Then we have the following formulation:

$$
\begin{aligned}
\text{minimize} & \sum_{(i,j)\in\mathcal{A}} \sum_{k=1}^{n} \sum_{\ell=1}^{n} c_{ij} x_{ij}^{k\ell} \\
\text{subject to} & \sum_{\{j \,|\, (i,j)\in\mathcal{A}\}} x_{ij}^{k\ell} - \sum_{\{j \,|\, (j,i)\in\mathcal{A}\}} x_{ji}^{k\ell} = b_i^{k\ell}, \quad i = 1, \ldots, n, k, \ell = 1, \ldots, n \\
& \sum_{k=1}^{n} \sum_{\ell=1}^{n} x_{ij}^{k\ell} \leq u_{ij}, \quad (i, j) \in \mathcal{A} \\
& x_{ij}^{k\ell} \geq 0, \quad (i, j) \in \mathcal{A}, k, \ell = 1, \ldots, n
\end{aligned}
$$

The first constraint is a flow conservation constraint at node i for data with origin k and destination ℓ. The expression:

$$
\sum_{\{j \,|\, (i,j)\in\mathcal{A}\}} x_{ij}^{k\ell}
$$

represents the amount of data with origin and destination k and ℓ, respectively, that leave node i along some link. The expression:

$$\sum_{\{j \,|\, (j,i) \in \mathcal{A}\}} x_{ji}^{k\ell}$$

represents the amount of data with the same origin and destination that enter node i through some link. Finally, $b_i^{k\ell}$ is the net amount of such data that enter node i from outside the network. The second constraint express the requirement that the total traffic through a link (i, j) cannot exceed the link's capacity.

We are given m examples of objects and for each one, say the ith one, a description of its features in terms of an n-dimensional vector a_i; More concretely, suppose that each object is an image of an apple or an orange (these are our two classes). We are interested in designing a classifier which, given a new object (other than the originally available examples) , will figure out whether it is an image of an apple or of an orange.

A linear calssifier is defined in terms of an n-dimensional vector $\mathbf{x}$ and a scalar x_{n+1}. Given a new object with feature vector $\mathbf{a}$, the classifier declares it to be an object of the first class and the second class if:

$$\mathbf{a'x \geq x_{n+1}} \quad \text{and} \quad \mathbf{a'x < x_{n+1}}$$

Let S be the set of examples of the first class. We are then looking for some $\mathbf{x}$ and x_{n+l} that satisfy the constraints

$$\mathbf{a_i'x \geq x_{n+1}}, \quad \mathbf{i \in S} \text{ and } \mathbf{a_i'x < x_{n+1}}, \quad \mathbf{i \notin S}$$

Note that the second set of constraints involves a strict inequality and is not quite of the form arising in linear programming. This issue can bebypassed by observing that if some choice of $\mathbf{x}$ and x_{n+1} satisfies all of the above constraints, then there exists some other choice that satisfies:

$$\mathbf{a_i'x \geq x_{n+1}}, \quad \mathbf{i \in S} \text{ and } \mathbf{a_i'x \leq x_{n+1}}, \quad \mathbf{i \notin S}$$

We conclude that the search for a linear classifier consistent with all available examples is a problem of finding a feasible solution to a linear programming problem.

1.2 Piecewise linear convex objective functions

Definition 1.2 (convex & concave).

- a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is called *convex* if for all $\mathbf{x}, \mathbf{y} \in \mathbb{R^n}$ and all $\lambda \in [0, 1]$, we have:

$$f(\lambda \mathbf{x} + (\mathbf{1} - \lambda)\mathbf{y}) \leq \lambda \mathbf{f}(\mathbf{x}) + (\mathbf{1} - \lambda)\mathbf{f}(\mathbf{y})$$

- a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is called *concave* if for all $\mathbf{x}, \mathbf{y} \in \mathbb{R^n}$ and all $\lambda \in [0, 1]$, we have:

$$f(\lambda \mathbf{x} + (\mathbf{1} - \lambda)\mathbf{y}) \geq \lambda \mathbf{f}(\mathbf{x}) + (\mathbf{1} - \lambda)\mathbf{f}(\mathbf{y})$$

Note that a function of the form $f(x) = a_0 + \sum_{i=1}^n a_i x_i$ where $a_0, \ldots, a_n$ are scalars, called an affine function, is both convex and concave. Let $\mathbf{c_1}, \ldots, \mathbf{c_m}$ be vectors in $\mathbb{R}^n$ and let $d_1, \ldots, d_m$ be scalars. The function $f : \mathbb{R}^n \mapsto \mathbb{R}$ defined by:

$$f(\mathbf{x}) = \max_{\mathbf{i=1,\ldots,m}} (\mathbf{c_i'x + d_i})$$

Theorem 1.1 (convexity of the maximum of convex functions). *Let $f_1, \ldots, f_m : \mathbb{R}^m \mapsto \mathbb{R}$ be convex functions. Then the function f defined by $f(\mathbf{x}) = \max_{\mathbf{i=1,\ldots,m}} \mathbf{f_i}(\mathbf{x})$ is convex.*

Proof. Let $\mathbf{x}, \mathbf{y} \in \mathbb{R}^{\mathbf{n}}$ and let $\lambda \in [0,1]$. We have:

$$\begin{aligned} f(\lambda\mathbf{x} + (\mathbf{1}-\lambda)\mathbf{y}) &= \max_{i=1,\ldots,m} f_i(\lambda\mathbf{x} + (\mathbf{1}-\lambda)\mathbf{y}) \\ &\leq \max_{i=1,\ldots,m} (\lambda f_i(\mathbf{x}) + (\mathbf{1}-\lambda)\mathbf{f_i}(\mathbf{y})) \\ &\leq \lambda \max_{i=1,\ldots,m} f_i(\mathbf{x}) + (\mathbf{1}-\lambda) \max_{\mathbf{i=1,\ldots,m}} \mathbf{f_i}(\mathbf{y}) \\ &= \lambda f(\mathbf{x}) + (\mathbf{1}-\lambda)\mathbf{f}(\mathbf{y}) \end{aligned}$$

□

A function of the form $\max_{i=1,\ldots,m}(\mathbf{c_i'x} + \mathbf{d_i})$ is called a piecewise linear convex function. We now consider a generalization of linear programming, where the objective function is piecewise linear and convex rather than linear:

$$\text{minimize} \max_{i=1,\ldots,m} (\mathbf{c_i'x} + \mathbf{d_i}) \quad \text{subject to} \mathbf{Ax} \geq \mathbf{b}$$

Example 1.3 (Problems involving absolute values)**.** Consider a problem of the form:

$$\begin{aligned} &\text{minimize} \sum_{i=1}^{n} c_i|x_i| \\ &\text{subject to} \mathbf{Ax} \geq \mathbf{b} \end{aligned}$$

where $\mathbf{x} = (\mathbf{x_1}, \ldots, \mathbf{x_n})$, where the cost coefficients c_i are assumed to be nonnegative. The cost criterion, being the sum of the piecewise linear convex functions $c_i|x_i|$ is easily shown to be piecewise linear and convex. However, expressing this cost criterion in the form $\max_j(\mathbf{c_j'x} + \mathbf{d_j})$ is not immediate. We observe that $|x_i|$ is the smallest number z_i that satisfies $x_i \leq z_i$ and $-x_i \leq z_i$, and we obtain the linear programming formulation:

$$\begin{aligned} &\text{minimize} \sum_{i=1}^{n} c_i z_i \\ &\text{subject to} \mathbf{Ax} \geq \mathbf{b} \\ &\qquad x_i \leq z_i, \quad i = 1, \ldots, n \\ &\qquad -x_i \leq z_i, \quad i = 1, \ldots, n \end{aligned}$$

An alternative method for dealing with absolute values is to introduce new varaibles x_i^+, x_i^-, constratined to be nonnegative and let $x_i = x_i^+ - x_i^-$. We then have:

$$\begin{aligned} &\text{minimize} \sum_{i=1}^{n} c_i(x_i^+ + x_i^-) \\ &\text{subject to} \mathbf{Ax}^+ - \mathbf{Ax}^- \geq \mathbf{b} \\ &\qquad \mathbf{x}^+, \mathbf{x}^- \geq 0 \end{aligned}$$

page 20

Part II

Theory of Linear and Integer Programming

Chapter 2

Linear Algebra and Complexity

2.1 Some Theory

A subset L of $\mathbb{R}^n$ ($\mathbb{Q}^n$) is a linear hyperplane if $L = \{x \big| ax = 0\}$ for some nonzero row vector $a \in \mathbb{R}$ (respectively $\mathbb{Q}^n$)

Theorem 2.1. *Linear Subspace Decomposition Each Linear subspace of $\mathbb{R}^n$ ($\mathbb{Q}^n$) is generated by finitely many vectors, and is also the intersection of finitely many linear hyperplanes.*

Corollary 2.2. *For each matrix A there exist vectors $x_1, \ldots, x_t$ such that: $Ax = 0$ if and only if $x = \lambda_1 x_1 + \ldots + \lambda_t x_t$ for certain rationals $\lambda_1, \ldots, \lambda_t$.*

Corollary 2.3 (Fundamental Theorem of Linear Algebra)**.** *The system $Ax = b$ has a solution if and only if $yb = 0$ for each vector y with $yA = 0$.*

Corollary 2.4 (inhomogeneous system)**.** *If x_0 is a solution of the system $Ax = b$, then there are vectors $x_1, \ldots, x_t$ such that $Ax = b$ if and only if $x = x_0 + \lambda_1 x_1 + \ldots + \lambda_t x_t$ for certain rationals $\lambda_1, \ldots, \lambda_t$ for cecrtain rationals $\lambda_1, \ldots, \lambda_t$.*

If $A = \begin{pmatrix} A_1 & A_2 \end{pmatrix}$ and A_1 is non-singular, then we can take $x_0, x_1, \ldots, x_t$ as in the above corollary so that:

$$\begin{pmatrix} x_0 & x_1 & \ldots & x_t \end{pmatrix} = \begin{pmatrix} A_1^{-1}b & A_1^{-1}A_2 \\ 0 & -I \end{pmatrix} \tag{2.1}$$

As for any nonsingular matrix C, the (j,i)th entry of C^{-1} is given by:

$$\frac{(-1)^{i+j} \det(C_{ij})}{\det C} \tag{2.2}$$

where C_{ij} arises from C by deleting the ith row and the jth column, it follows that for any given column vector d, the jth component of $C^{-1}d$ is given by:

$$\frac{\det \tilde{C}}{\det C} \tag{2.3}$$

where $\tilde{C}$ arises from C by replacing the jth column by d. This is the Cramer's rule.

Corollary 2.5. *Let A be a matrix, b be a column vector, c be a row vector, and δ be a number. Assume that the system $Ax = b$ has a solution, then $Ax = b$ implies $cx = \delta$ if and only if there is a row vector y such that $yA = c$ and $yb = \delta$.*

Proof. The sufficiency of the condition is trivial. To see necessity, let x_0 be a solution of $Ax = b$ and suppose $Ax = b$ implies $cx = \delta$. Then in particular, $cx_0 = \delta$. Then: $Ax = 0 \implies A(x + x_0) = b \implies c(x + x_0) = \delta \implies cx = 0$. Therefore, by corollary, we have $yA = c$ for some row vector y. This y also satisfies $yb = yAx_0 = cx_0 = \delta$. □

2.2 Sizes and Good Characterizations

We use the following as the sizes of a rational number $r = p/q$ ($p \in \mathbb{Z}, q \in \mathbb{N}, p, q$ are relative prime) of a rational vector $c = (\gamma_1, \ldots, \gamma_n)$ and of a rational matrix $A = (\alpha_{i=1,j=1}^{m,n})$:

$$\begin{aligned} \text{size}(r) &= 1 + \lceil \log_2(|p| + 1) \rceil + \lceil \log_2(q + 1) \rceil \\ \text{size}(c) &= n + \sum_{i=1}^{n} \text{size}(\gamma_i) \\ \text{size}(A) &= mn + \sum_{i=1}^{m} \sum_{j=1}^{n} \text{size}(\alpha_{ij}) \end{aligned} \tag{2.4}$$

Theorem 2.6 (Size of Determinant of a Matrix)**.** *Let A be a square rational matrix of size σ, then the size of $\det A$ is less than 2σ.*

Proof. Let $A = (p_{ij}/q_{ij})_{i,j=1}^{n}$ where for each i, j, p_{ij}, q_{ij} are relative prime integers where $q_{ij} > 0$. Moreover, let $\det A = p/q$ where p and q are relatively prime integers, with $q > 0$. Clearly,

$$q \leq \Pi_{i,j=1}^{n} q_{ij} < 2^{\sigma - 1} \tag{2.5}$$

It is immediate from the definition of a determinant that:

$$|\det A| \leq \Pi_{i=1}^{n} \Pi_{j=1}^{n} (|p_{ij}| + 1) \tag{2.6}$$

Combining the equations together we have

$$|p| = |\det A| \cdot q \leq \Pi_{i,j=1}^{n} (|p_{ij}| + 1) q_{ij} < 2^{\sigma - 1} \tag{2.7}$$

Therefore, we have

$$\text{size}(\det A) = 1 + \lceil \log_2(|p| + 1) \rceil + \lceil \log_2(q + 1) \rceil < 2\sigma \tag{2.8}$$

□

Corollary 2.7. *The inverse A^{-1} of a nonsingular rational matrix A has size polynomially bounded by the size of A.*

Corollary 2.8. *If the system $Ax = b$ of rational linear equations has a solution, it has one of size polynomially bounded by the size of A and b.*

Proof. we may assue that A has linear independent rows, and that $A = \begin{pmatrix} A_1 & A_2 \end{pmatrix}$ with A_1 nonsingular. Then by the previous corollary, we have:

$$x_0 \equiv \begin{pmatrix} A_1^{-1} b \\ 0 \end{pmatrix} \tag{2.9}$$

is a solution of $Ax = b$ of size polynomially bounded by the size of A and b. □

Corollary 2.9. *Let A be a rational $m \times n$ matrix and let b be a rational column vector such that each row of the matrix $\begin{pmatrix} A & b \end{pmatrix}$ has size at most φ. If $Ax = b$ has a solution, then:*

$$\{x \big| Ax = b\} = \{x_0 + \lambda_1 x_1 + \ldots + \lambda_t x_t \big| \lambda_1, \ldots, \lambda_t \in \mathbb{R}\} \tag{2.10}$$

for certain rational vectors $x_0, \ldots, x_t$ of size at most $4n^2\varphi$.

Proof. By Cramer's rule, there are $x_0, \ldots, x_t$ satisfying with all nonzero components being quotients of subdeterminants of the matrix $\begin{pmatrix} A & b \end{pmatrix}$ of order at most n. Therefore, it has size less than $2n\varphi$. Hence, each component of x_t has size less than $4n\varphi$. □

2.3 Gaussian Elimination Method

Suppose we want to solve the system:

$$\begin{aligned} \alpha_{11}x_1 + \ldots + \alpha_{1n}x_n &= \beta_1 \\ &\vdots \\ \alpha_{m1}x_1 + \ldots + \alpha_{mn}x_n &= \beta_m \end{aligned} \tag{2.11}$$

We may suppose $\alpha_{11} \neq 0$. Then we subtract approximation multiples of the first equation from the other equations so to get and recursively to solve the equations:

$$\begin{aligned} \alpha'_{11}x_1 + \ldots + \alpha'_{1n}x_n &= \beta_1 \\ \alpha'_{21}x_2 + \ldots + \alpha'_{2n}x_n &= \beta'_2 \\ &\vdots \\ \alpha'_{nn}x_n &= \beta'_m \end{aligned} \tag{2.12}$$

IN matrix form, the Gaussian Elimination method transform a given matrix A into the form:

$$\begin{pmatrix} B & C \\ 0 & 0 \end{pmatrix} \tag{2.13}$$

where B is a nonsingular, upper triangle matrix.

Theorem 2.10 (Gaussian Elimination Method Running Time)**.** *For rational data, the Gaussian Elimination method is a polynomial time method.*

Proof. Without loss of generality, we may assume that we do not need to permute rows or columns at executing the method. It is moreover easy to see that the number of elementary arithmetric operations like addition, multiplication and division, in the method is polynomially bounded by the number of rows and columns of the initial matrix A, and hence by size(A). So to prove that the method is polynomial, it is only left to show that the sizes of the numbers occuring throughout the method are polynomially bounded by size(A).

To prove this, note that A_k and δ_{ij} is the (i,j)th entry of the matrix D, then trivially:

$$\delta_{ij} = \det\Big((A_k)^{1,\ldots,k,k+i}_{1,\ldots,k,k+j}\Big) / \det\Big((A_k)^{1,\ldots,k}_{1,\ldots,k}\Big) \tag{2.14}$$

Here $G^{i_1,\ldots,i_k}_{j_1,\ldots,j_k}$ denotes the submatrix of G induced by rows $i_1, \ldots, i_k$ and columns $j_1, \ldots, j_k$. As A_k rises from A by adding multiples of the first k rows to other rows, and the result also holds if we replace A_k by A, i.e.,

$$\delta_{ij} = \det\Big((A)^{1,\ldots,k,k+i}_{1,\ldots,k,k+j}\Big) / \det\Big((A)^{1,\ldots,k}_{1,\ldots,k}\Big) \tag{2.15}$$

Therefore, the sie of δ_{ij} is at most $4\text{size}(A)$. To see that also during the backward step the entries do not grow too large in size, let E be the matrix formed by applying for $k = r, r-1, \ldots, t+1$ SO:

$$E = \begin{pmatrix} B & 0 & C \\ 0 & \Delta & D \\ 0 & 0 & 0 \end{pmatrix} \tag{2.16}$$

where B is a nonsingular upper-triangle matrix and Δ is a diagonal $(r-t) \times (r-t)$ matrix. Spliting A_r accordingly, we have:

$$A_r = \begin{pmatrix} B_1 & B_2 & C_1 \\ 0 & B_3 & C_2 \\ 0 & 0 & 0 \end{pmatrix} \tag{2.17}$$

with B_1 and B_3 upper-traingle matrices of order t and $r-t$ respectively. Then one easily checks that $B = B_1, C = C_1 - B_2 B_3^{-1} C_2, D = \Delta B_3^{-1} C_2$ and Δ consists of the diagonal elements of B_3. Therefore, the size of E is polynomially bounded by size(A_r) and hence by size(A). □

Corollary 2.11. *The following problems are polynomially solvable:*

- *determining the determinant of a rational matrix*
- *determining the rank of rational matrix*
- *determining the inverse of a nonsingular rational matrix*
- *testing rational vectors for linear independence*
- *solving a system of rational lienar equations*

Chapter 3

Theory of lattices and linear diophantine equations

3.1 The Hermite Normla Form

A matrix of full row rank is said to be in Hermite normal form if it has the form $\begin{pmatrix} B & 0 \end{pmatrix}$ where B is a nonsingular lower triangle, nonnegative matrix, in which each row has a unique maximum entry, which is located on the main diagonal of B.

The following operations on a matrix are called elementary (unimodular) column operations:

- exchanging two columns;
- multiplying a column by -1;
- adding an integral multiple of one column to another column;

Theorem 3.1 (Hermite normal form theorem)**.** *Each rational matrix of full row rank can be brought into Hermite normal form by a series of elementary column operations.*

Proof. Let A be a rational matrix of full row rank. Without loss of generality, A is integral. Suppose we have transformed A, by elementary column operations, to the form $\begin{pmatrix} B & 0 \\ C & D \end{pmatrix}$ where B is lower triangle and with positive diagonal.

Now with elementary column operations we can modify D so that its first row $(\delta_{11}, \ldots, \delta_{1k})$ is nonnegative, and so that the sum $\delta_{11} + \ldots + \delta_{1k}$ is as small as possible. We may assume that $\delta_{11} \geq \delta_{12} \geq \ldots \geq \delta_{1k}$. Then $\delta_{11} > 0$ as A has full row rank. Moreover, if $\delta_{12} > 0$, by subtracting the second column of D from the first column of D, the first row will have smaller sum, contradicting our assumption.

By repeating the procedure, the matrix A finally will be transformed into $\begin{pmatrix} B & 0 \end{pmatrix}$ with $B = (\beta_{ij})$ lower triangle with positive diagonal. Next, we do the following:

for $i = 2, \ldots, n$, do the following: for $j = 1, \ldots, i-1$, add an integer multiple of the ith column of B to the jth column of B so that the (i, j)th entry of B will be nonnegative and less than β_{ii}. □

Corollary 3.2 (Kronecker's approximation theorem)**.** *Let A be a rational matrix and let b be a rational column vector. Then the system $Ax = b$ has an integer solution x if and only if yb is an integer for each rational row vector y for whcih yA is integer.*

Proof. Necessity of the condition is trivial: if x and yA are integer vectors and $Ax = b$, then $yA = yAx$ is an integer.

To see sufficiency, suppose yb is an integer whenever yA is integer. Then $Ax = b$ has a (possibly fractional) solution, since otherwise $yA = 0$ and $yb = \frac{1}{2}$ for some rational vector y. So we may assume that the rows of A are linearly independent. Now both sides of the equivalence to be proved are invariant under elementary column operations. Therefore, by the Theorem, we may assume that A is in Hermite normal form $\begin{pmatrix} B & 0 \end{pmatrix}$. Since $B^{-1}\begin{pmatrix} B & 0 \end{pmatrix} = \begin{pmatrix} I & 0 \end{pmatrix}$ is an integral matrix, it follows from our assumption that also $B^{-1}b$ is an integer vector. Since:

$$\begin{pmatrix} B & 0 \end{pmatrix}\begin{pmatrix} B^1 b \\ 0 \end{pmatrix} = b \tag{3.1}$$

the vector $x := \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$ is an integer solution for $Ax = b$. □

A subset $\wedge$ of $\mathbb{R}^n$ is called an (additive) group if:

- $0 \in \wedge$;
- if $x, y \in \wedge$ then $x + y \in \wedge$ and $-x \in \wedge$;

The group is said to be generated by $a_1, \ldots, a_m$ if:

$$\wedge = \{\lambda_1 a_1 + \cdots + \lambda_m a_m \big| \lambda_1, \ldots, \lambda_m \in \mathbb{Z}\} \tag{3.2}$$

The group is called a lattice if it can be generated by the linearly independent vectors. These vectors then are called a basis for the lattice.

Corollary 3.3. *If $a_1, \ldots, a_m$ are rational vectors, then the group generated by $a_1, \ldots, a_m$ is a lattice, i.e., it is generated by linearly independent vectors.*

Proof. We may assume that $a_1, \ldots, a_m$ span all space. Let A be the matrix with columns $a_1, \ldots, a_m$ (so A has full row rank). Let $\begin{pmatrix} B & 0 \end{pmatrix}$ be the Hermite normal form of A. Then the columns of B are linearly independent vectors generating the same group as $a_1, \ldots, a_m$. □

Corollary 3.4. *Let A be an integral $m \times n$ matrix of full row rank. Then the following are equivalent:*

- *the g.c.d. of the subdeterminants of A of order m is 1;*
- *the system $Ax = b$ has an integral solution x, for each integral vector b;*
- *for each vector y, if yA is integral then y is integral.*

It can be derived easily from the Hermite normal form theorem that for any rational system $Ax = b$ with at least ne integral solution there exist integral vectros $x_0, x_1, \ldots, x_t$ such that:

$$\{x \big| Ax = b; x \text{ integral}\} = \{x_0 + \lambda_1 x_1 + \cdots + \lambda_t x_t \big| \lambda_1, \ldots, \lambda_t \in \mathbb{Z}\} \tag{3.3}$$

where $x_1, \ldots, x_t$ are linearly independent and $t =$ (number of columns of A) - rank(A) The existence of such a system of fundamental solutions are stated in Smith.

3.2 Uniqueness of the Hermite Normal Form

Theorem 3.5 (Uniqueness of the Hermite Normal Form). *Let A and A' be rational matrices of full row rank, with Hermite normal form $\begin{pmatrix} B & 0 \end{pmatrix}$ and $\begin{pmatrix} B' & 0 \end{pmatrix}$, respectively. Then the columns of A generate the same lattice as those of A', if and only if $B = B'$.*

Proof. Sufficiency is clear, as the columns of B and A generate the same lattice, and similarly for B' and A'.

To see necessity, suppose the columns of A and those of A' generate the same lattice $\wedge$. Then the same holds for B and B', as they come by elementary column operations from A and A'. Write $\beta = (\beta_{ij})$ and $B' = (\beta'_{ij})$. Suppose $B \neq B'$ and choose $\beta_{ij} \neq \beta'_{ij}$ wiht i as small as possible. WLOG, $\beta_{ii} \geq \beta'_{ii}$. Let b_j and b'_j be the jth columns of B and B'. Then $b_j \in \wedge$ and $b'_j \in \wedge$, and hence, $b_j - b'_j \in \wedge$. This implies that $b_j - b'_j$ is an integral linear combination of the columns of B. By our choice of i, the vector $b_j - b'_j$ has zeros in the first $i-1$ positions. Hence, as B is lower triagnle, $b_j - b'_j$ is an integral linear combination of the column indexed $i, \ldots, n$. So $\beta_{ij} - \beta'_{ij}$ is an integral multiple of β_{ii}. However, this contradicts the fact that $0 < |\beta_{ij} - \beta_{ii}$ and $j < i$ then $0 \leq \beta_{ij} < \beta_{ii}$ and $0 < \beta'_{ij} < \beta'_{ii} \leq \beta_{ii}$. □

Corollary 3.6. *Every rational matrix of full row rank has a uinque Hermite normal form.*

3.3 Unimodular Matrices

Series of elementary column operations can be described by so-called unimodular matrices. let U be a non-singular matrix and then U is called unimodular if U is integral and has determinant ± 1.

Theorem 3.7 (unimodular and ralations). *The following are equivalent for a nonsingular ratoinal matrix U of order n:*

- *U is unimodular*
- *U^{-1} is unimodular*
- *the lattice generated by the columns of U is $\mathbb{Z}^n$*
- *U has the identity matrix as its Hermite normal form;*
- *U comes from the identity matrix by elementary column operations.*

Proof. (i) implies (ii) as $\det(U^{-1}) = (\det U)^{-1} = \pm 1$ and as each entry of U^{-1} is a subdeterminant of U, and hence is an integer. Similarly, (ii) implies (i).

The equivalence of (iii), (iv) and (v) follows directly from Theorem above, and the implication (v) $\implies$ (i) is trivial. □

Corollary 3.8. *Let A and A' be nonsingular matrices. Then the following are equivalent:*

- *The columns of A and those of A' generate the same lattice;*
- *A' comes from A by elementary column operations;*
- *$A' = AU$ for some unimodular matrix U (i.e., $A^{-1}A'$ is unimodular).*

Corollary 3.9. *For each rational matrix A f full row rank there is a unimodular matrix U such that AU is the Hermite normal form of A. If A is nonsingular, U is unique.*

3.4 Future Rmarks

If $\wedge$ if a full-dimensional lattice, defined the dual lattice $\wedge^{\perp}$ by:

$$\wedge^{\perp} = \{z \big| zx \text{ is an integer for all } x \text{ in } \wedge\} \tag{3.4}$$

If $\wedge$ is generated by the columns of the nonsingular matrix B, then $\wedge^{\perp}$ is the lattice generated by the rows of B^{-1}. and $\wedge^{\perp\perp} = \wedge$.

Chapter 4

Algorithms for Linear Diophantine equations

4.1 The Euclidean Algorithm

The Euclidean algorihtm determines, in polynomial time, the g.c.d. of tow positive rational numbers α and β. First replace α by $\alpha - \lfloor \alpha/\beta \rfloor \beta$ and β is replaced by $\beta - \lfloor \beta/\alpha \rfloor \alpha$. Next this is repeated until one of α and β is 0. In this case, the nonzero among α and β is the g.c.d. of the original α and β. This follows from the facts that g.c.d. $\{\alpha, \beta\} = g.c.d.\{\alpha, \lfloor \alpha/\beta \rfloor, \beta\}$ and $g.c.d.\{\alpha, 0\} = \alpha$.

Let two positive rationals α and β. Determine a series of 3×2-matrices $A_0, A_1, A_2, \ldots$ as follows. The matrix A_0 is given by:

$$A_0 := \begin{pmatrix} \alpha & \beta \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{4.1}$$

If A_k has been found, say:

$$A_k = \begin{pmatrix} \alpha_k & \beta_k \\ \gamma_k & \delta_k \\ \varepsilon_k & \xi_k \end{pmatrix} \tag{4.2}$$

Let A_{k+1} arise from A_k by the following rule:
If A_{k+1} arise from A_k by the following rule:

- if k is even and $\beta_k > 0$ subtract $\lfloor \alpha_k/\beta_k \rfloor$ times the second column of A_k from the first column;
- If k is odd and α_k subtract $\lfloor \beta_k/\alpha_k \rfloor$ times the first column of A_k from the second column.

This step is performed for $k = 0, 1, 2, \ldots$ until $k = N$ (say) $\alpha_N = 0, \beta_N = 0$. If $\beta_N = 0$ and $\alpha_N \neq 0$ then α_N is the g.c.d. of α and β, as during the iterations the g.c.d. of the entries in the upper row of A_k does not change, and as $\alpha_N = g.c.d.\{\alpha_N, 0\}$. Similarly, if $\alpha_N = 0, \beta_N \neq 0$ then $\beta_N = g.c.d.\{\alpha, \beta\}$.

We also find integers γ and ε with $\gamma\alpha + \varepsilon\beta = g.c.d.\{\alpha, \beta\}$. To see this, observe that $(1, -\alpha, -\beta)A_0 = (0, 0)$ and hence, for each k, $(1, -\alpha, -\beta)A_k = (0, 0)$ as we did only elementary column operations. In the notation above, this means:

$$\gamma_k\alpha + \varepsilon_k\beta = \alpha_k \text{ and } \delta_k\alpha + \xi_k\beta = \beta_k \tag{4.3}$$

Therefore, if $\beta_N = 0, \alpha_N \neq 0$ then we have:

$$\gamma_N\alpha + \varepsilon_N\beta = \alpha_N = g.c.d.\{\alpha,\beta\} \text{ and } \delta_N\alpha + \xi_N\beta = \beta = 0 \tag{4.4}$$

Similarly for $\alpha_N = 0, \beta_N \neq 0$.

The Euclidean algorithm indeed terminates with polynomial time. We observe that for each k,

$$-\alpha_k\delta_k + \beta_k\gamma_k = \beta \text{ and } \alpha_k\xi_k - \beta_k\varepsilon_k = \alpha \tag{4.5}$$

Theorem 4.1 (Euclidean Algorithm Terminates in Polynomial Time)**.** *The Euclidean algorithm terminates in polynomial time.*

Proof. We may assume that α and β are natural numbers. So all matrices A_k have nonnegative integers in their first row. Since at each iteration either α_k or β_k is reduced by a factor of at least 2, after at most $\lfloor \log_2 \alpha \rfloor + \lfloor \log_2 \beta \rfloor + 1$ iterations, one of α_k and β_k is 0.

Now for each k: $\alpha_k \leq \alpha, \beta_k \leq \beta$. It follows that $|\delta_k| \leq \beta, |\gamma_k| \leq \beta, |\xi_k| \leq \alpha, |\varepsilon_k| \leq \alpha$ as long as $\alpha_k > 0, \beta_k > 0$. This shows that throughout the method the sizes of the numbers are polynomially bounded. □

Corollary 4.2. *A linear diophantine equaiton with rational coefficients can be solved in polynomial time.*

Proof. Let

$$\alpha_1\xi_1 + \cdots + \alpha_n\xi_n = \beta \tag{4.6}$$

be a rational linear diophantine equaiton. The algorithm solves the above equation is described recursively on n. $n = 1$ is trivial. Let $n \geq 2$ and find with the Euclidean algorihtm $\alpha', \gamma, \varepsilon$ satisfying:

$$\alpha' = g.c.d.\{\alpha_1,\alpha_2\} \text{ and } \alpha_1\gamma + \alpha_2\varepsilon = \alpha', \gamma, \varepsilon \in \mathbb{Z} \tag{4.7}$$

Next solve the linear diophantine equation:

$$\alpha'\xi_1 + \alpha'_3\xi_3 + \cdots + \alpha_n\xi_n = \beta \tag{4.8}$$

If the above euqaiton has no integral solution, then neither has the original problem, and if the above equation has an integral solution, then the original problem has an integral solution. The Euclidean algorithm terminates in polynomial time, and hence the above recursive algorithm terminates in polynomial time. □

4.2 sizes and good characterizations

Theorem 4.3 (good characterizations)**.** *tabularxhe Hermite normal form* $\begin{pmatrix} B & 0 \end{pmatrix}$ *of a rational matrix* A *of full row rank has size polynomially bounded by the size of* A*. Moreover, there exists a unimodular matrix* U *with* $AU = \begin{pmatrix} B & 0 \end{pmatrix}$ *such that the size of* U *is polynomially bounded by the size of* A*.*

Proof. We may assume that A is integral, as multiplying A by the product, say κ, of the denominators occuring in A also multiples the Hermite normal form of A by κ.

The diagonal entries of B are divisors of subdeterminants of A. As each row of B has its maximum entry on the main diagonal of B, it follows that the size of $\begin{pmatrix} B & 0 \end{pmatrix}$ is polynomially bounded by the size of A.

By permuting columns, we may assume that $A = \begin{pmatrix} A' & A'' \end{pmatrix}$ with A' is non-singular. The Hermite normal form of the matrix:

$$\begin{pmatrix} A' & A'' \\ 0 & I \end{pmatrix} \text{ is } \begin{pmatrix} B & 0 \\ B' & B'' \end{pmatrix} \tag{4.9}$$

for certain B' and B''. As the sizes of B, B', B'' are polynomially bounded by the size of A, the size of the unimodular matrix U is polynomially bounded by the size of A.

$$U := \begin{pmatrix} U' & U'' \\ 0 & I \end{pmatrix}^{-1} \begin{pmatrix} B & 0 \\ B' & B'' \end{pmatrix} \tag{4.10}$$

is polynomially bounded by the size of A. Now $AU = \begin{pmatrix} A' & A'' \end{pmatrix} U = \begin{pmatrix} B & 0 \end{pmatrix}$. □

Corollary 4.4. *If a rational system $Ax = b$ has an integral solution, it has one of size polynomially bounded by the size of A and b.*

Proof. WLOG, A has full row rank. Let $\begin{pmatrix} B & 0 \end{pmatrix} = AU$ be the Hermite normal form of A, with U unimodular of size polynomially bounded by the size of A. Now $B^{-1}b$ is integral. Therefore,

$$\tilde{x} := U \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix} \tag{4.11}$$

is an integral solution of $Ax = b$ of size polynomially bounded by the size of A and b. □

Corollary 4.5. *The following problem has a good characterization: given a rational matrix A and a rational vector b, does the system $Ax = b$ has an integral solution?*

Proof. The general case can be easily reduced to the case where A has full row rank.

If the answer is positive there is a solution of polynomial size, and if it is negative, the there exists a rational row vector y with yA integral and yb not a integer. WE can take y of polynomially bounded size: since $B^{-1}A$ is integral but $B^{-1}b$ is not, we can take for y one of the rows B^{-1}. □

4.3 Polynomial Algorithms for Hermite Normal Forms and Systems of Linear Diophantine Equations

Let an integral $m \times n$ matrix A, of full rank, be given. Let M be the absolute value of the determinant of an (arbitrary) submatrix of A of rank m. Now the columns of A generate the same lattice as the columns of the matrix:

$$A' = \begin{pmatrix} A \middle| M \end{pmatrix} \tag{4.12}$$

Part III

Homework and Solution