Optimization and Approximation

Elisa Riccietti[1][2]

[1]Thanks to Stefania Bellavia, University of Florence.
[2]Reference book: Numerical Optimization, Nocedal and Wright, Springer

Contents

I I part: nonlinear optimization **5**

1 Prerequisites **7**
1.1 Necessary and sufficient conditions . . . 9
1.2 Convex functions . . . 10
1.3 Quadratic functions . . . 11

2 Iterative methods **13**
2.1 Directions for line-search methods . . . 14
2.1.1 Direction of steepest descent . . . 14
2.1.2 Newton's direction . . . 15
2.1.3 Quasi-Newton directions . . . 16
2.2 Rates of convergence . . . 16
2.3 Steepest descent method for quadratic functions . . . 17
2.4 Convergence of Newton's method . . . 19

3 Line-search methods **23**
3.1 Armijo and Wolfe conditions . . . 23
3.2 Convergence of line-search methods . . . 27
3.3 Backtracking . . . 30
3.4 Newton's method . . . 32

4 Quasi-Newton method **33**
4.1 BFGS method . . . 34
4.2 Global convergence of the BFGS method . . . 38

5 Nonlinear least-squares problems **41**
5.1 Background: modelling, regression . . . 41
5.2 General concepts . . . 41
5.3 Linear least-squares problems . . . 43
5.4 Algorithms for nonlinear least-squares problems . . . 44
5.4.1 Gauss-Newton method . . . 44
5.5 Levenberg-Marquardt method . . . 45

6 Constrained optimization **47**
6.1 One equality constraint . . . 48
6.2 One inequality constraint . . . 50
6.3 First order optimality conditions . . . 52

6.4 Second order optimality conditions 58

7 Optimization methods for Machine Learning 61

II Linear and integer programming 63

8 Linear programming 65
8.1 How to rewrite an LP in standard form 66
8.2 Primal and dual problems 69
8.3 Convex and strictly convex problems 71
8.4 Geometry of Ω 72
8.5 Simplex method 77
8.5.1 How to choose a starting vertex of Ω 84
8.5.2 Generalization of the algorithm to the degenerate case 85
8.5.3 Advantages and disadvantages of the simplex method 87

9 Flow networks problems 89
9.1 Minimum cost flow problem 89
9.2 Maximum flow problem 91
9.2.1 How to find a starting point of Ω 99

Part I

I part: nonlinear optimization

Chapter 1

Prerequisites

Let A be an open set of $\mathbb{R}^n$ and let

$$\begin{array}{rccc} f: & A \subseteq \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & \mathbf{x} = (x_1, \ldots, x_n)^T & \longmapsto & f(\mathbf{x}) \end{array}$$

An unconstrained optimization problem is a problem of the form

$$\min_x f(x),$$

where f is called the *objective function*. In the following we will assume f to be a nonlinear function.

1. If f is differentiable in $\mathbf{x}$ (i.e. if there exist all the partial derivatives of f in $\mathbf{x}$), the *gradient* of f in $\mathbf{x}$ is $\nabla f(\mathbf{x}) \in \mathbb{R}^n$:

$$\nabla f(\mathbf{x}) = \begin{pmatrix} \dfrac{\partial f(\mathbf{x})}{\partial x_1} \\ \vdots \\ \dfrac{\partial f(\mathbf{x})}{\partial x_n} \end{pmatrix}.$$

2. If f is two times differentiable in $\mathbf{x}$, the *Hessian* matrix of f in x is $H(\mathbf{x}) \in \mathbb{R}^{n \times n}$

$$H(\mathbf{x}) = H_f(\mathbf{x}) = \begin{pmatrix} \dfrac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_1} & \cdots & \dfrac{\partial^2 f(\mathbf{x})}{\partial x_1 \partial x_n} \\ \vdots & & \vdots \\ \dfrac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_1} & \cdots & \dfrac{\partial^2 f(\mathbf{x})}{\partial x_n \partial x_n} \end{pmatrix} = \begin{pmatrix} \left(\nabla \dfrac{\partial f(\mathbf{x})}{\partial x_1}\right)^T \\ \vdots \\ \left(\nabla \dfrac{\partial f(\mathbf{x})}{\partial x_n}\right)^T \end{pmatrix}.$$

   If $f \in C^2(\mathbf{x})$ then $H(\mathbf{x})$ is a symmetric matrix.

3. Let us remind the *first-order Taylor formula* with Lagrange form of the remainder. Let $f \in C^1(A)$. Let $\mathbf{x}, \mathbf{x}+\mathbf{h} \in A$ with $\mathbf{h} \neq \mathbf{0}$ such that the segment $\{\mathbf{x}+t\mathbf{h} \mid t \in [0,1]\}$ whose endpoints are $\mathbf{x}$ and $\mathbf{x}+\mathbf{h}$ is contained in A. Then it exists $t \in (0,1)$, depending on $\mathbf{x}$ and $\mathbf{h}$, such that

$$f(\mathbf{x}+\mathbf{h}) = f(\mathbf{x}) + \nabla f(\mathbf{x}+t\mathbf{h})^T \mathbf{h}. \tag{1.1}$$

4. Let us remind the *second-order Taylor formula* with Lagrange form of the remainder. Let $f \in C^2(A)$. Let $\mathbf{x}, \mathbf{x}+\mathbf{h} \in A$ with $\mathbf{h} \neq \mathbf{0}$ such that the segment $\{\mathbf{x}+t\mathbf{h} \mid t \in [0,1]\}$ whose endpoints are $\mathbf{x}$ and $\mathbf{x}+\mathbf{h}$ is contained in A. Then it exists $t \in (0,1)$, depending on $\mathbf{x}$ and $\mathbf{h}$, such that

$$f(\mathbf{x}+\mathbf{h}) = f(\mathbf{x}) + \nabla f(\mathbf{x})^T \mathbf{h} + \frac{1}{2}\mathbf{h}^T H(\mathbf{x}+t\mathbf{h})\mathbf{h}. \tag{1.2}$$

5. f is *convex* in A if $\forall \mathbf{x}, \mathbf{y} \in A$

$$f(t\mathbf{x}+(1-t)\mathbf{y}) \leq tf(\mathbf{x}) + (1-t)f(\mathbf{y}), \quad \forall t \in [0,1].$$

6. f is *strictly convex* in A if $\forall \mathbf{x}, \mathbf{y} \in A$

$$f(t\mathbf{x}+(1-t)\mathbf{y}) < tf(\mathbf{x}) + (1-t)f(\mathbf{y}), \quad \forall t \in (0,1).$$

Definition 1.0.1. *Let $\mathbf{x}^* \in A$.*

- $\mathbf{x}^*$ *is a local minimizer or a local minimum point for f if it exists a neighbourhood Ω of $\mathbf{x}^*$ such that*
$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \qquad \forall \mathbf{x} \in \Omega.$$
$f(x^*)$ is a local minimum of f.
- $\mathbf{x}^*$ *is a global minimizer or a global minimum point for f if*
$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \qquad \forall \mathbf{x} \in A.$$
$f(x^*)$ is a global minimum of f.
- $\mathbf{x}^*$ *is a stationary point for f if*
$$\nabla f(\mathbf{x}^*) = 0.$$

Definition 1.0.2. *Directional derivatives*
Let $\mathbf{p} \in \mathbb{R}^n$ and f differentiable in a neighbourhood of $\mathbf{x}$. The directional derivative of f in $\mathbf{x}$ with respect to the direction $\mathbf{p}$ is defined as

$$\frac{\partial f}{\partial \mathbf{p}}(\mathbf{x}) = \lim_{h \to 0} \frac{f(\mathbf{x}+h\mathbf{p}) - f(\mathbf{x})}{h}. \tag{1.3}$$

It holds (see TD1)

$$\frac{\partial f}{\partial \mathbf{p}}(\mathbf{x}) = \nabla f(\mathbf{x})^T \mathbf{p}.$$

Definition 1.0.3. *Descent directions*
A direction $\mathbf{p} \in \mathbb{R}^n$ is a descent direction for f in $\mathbf{x}$ if

$$\frac{\partial f}{\partial \mathbf{p}}(\mathbf{x}) = \nabla f(\mathbf{x})^T \mathbf{p} < 0,$$

i.e., if the angle ϑ between $\mathbf{p}$ and $\nabla f(\mathbf{x})$ is such that $\vartheta \in \left(\frac{\pi}{2}, \pi\right]$.

If $\nabla f(\mathbf{x}) \neq \mathbf{0}$, we can always find a descent direction: that of the antigradient $-\nabla f(\mathbf{x})$. From (1.3) it follows that if $\mathbf{p}$ is a descent direction, then it exists $\bar{h} > 0$ such that

$$f(\mathbf{x}+h\mathbf{p}) - f(\mathbf{x}) < 0 \quad \forall h \in (0, \bar{h}).$$

1.1 Necessary and sufficient conditions

In this section we give necessary and sufficient conditions for a point to be a minimum point.

Theorem 1.1.1. *First order necessary condition*
Let $f \in C^1(\Omega)$ in a neighbourhood Ω of $\mathbf{x}^*$.

$$\mathbf{x}^* \text{ is a minimizer for } f \text{ (in } \Omega) \implies \nabla f(\mathbf{x}^*) = \mathbf{0}, \text{ i.e. } \mathbf{x}^* \text{ is a stationary point for } f.$$

Proof. We prove it by contradiction. Let us assume that $\nabla f(\mathbf{x}^*) \neq \mathbf{0}$. Let

$$\mathbf{p} = -\nabla f(\mathbf{x}^*)$$

the antigradient of f in $\mathbf{x}^*$, clearly $\mathbf{p} \neq \mathbf{0}$. The function

$$g(\mathbf{x}) = \nabla f(\mathbf{x})^T \mathbf{p}$$

is such that

$$g(\mathbf{x}^*) = \nabla f(\mathbf{x}^*)^T \mathbf{p} = -\nabla f(\mathbf{x}^*)^T \nabla f(\mathbf{x}^*) = -\|\nabla f(\mathbf{x}^*)\|^2 < 0.$$

Then, as $f \in C^1(\Omega)$ and so g is continuous in Ω, it will remain negative in a neighbourhood of x^*, i.e., it exists $T \in \mathbb{R}, T > 0$ such that $\forall t \in [0, T]$

$$0 > g(\mathbf{x}^* + t\mathbf{p}) = \nabla f(\mathbf{x}^* + t\mathbf{p})^T \mathbf{p}. \tag{1.4}$$

From (1.1), $\forall \tau \in (0, T)$, it exists $t \in (0, 1)$ such that

$$f(\mathbf{x}^*+\tau\mathbf{p}) = f(\mathbf{x}^*)+\nabla f(\mathbf{x}^*+ \underbrace{t\tau}_{=t' \in (0,T)} \mathbf{p})^T \tau\mathbf{p} = f(\mathbf{x}^*)+\tau \underbrace{\nabla f(\mathbf{x}^* + t'\mathbf{p})^T \mathbf{p}}_{<0 \text{ from (1.4)}} < f(\mathbf{x}^*).$$

Then $\mathbf{x}^*$ cannot be a minimum point for f, which leads us to a contradiction. □

This is just a necessary conditions, all minimizers are stationary points but not all stationary points are minimizers, they may be maximizers or saddle points.

Theorem 1.1.2. *Second order necessary condition*
Let $f \in C^2(\Omega)$ for a neighbourhood Ω of $\mathbf{x}^*$.

$$\mathbf{x}^* \text{ is a minimum point for } f \text{ (in } \Omega) \implies H(\mathbf{x}^*) \text{ is positive semidefinite.}$$

Proof. We do the proof by contradiction. Let us assume that $H(\mathbf{x}^*)$ is not positive semidefinite, i.e. that it exists $\mathbf{p} \in \mathbb{R}^n, \mathbf{p} \neq \mathbf{0}$ such that $\mathbf{p}^T H(\mathbf{x}^*)\mathbf{p} < 0$. Let us define

$$g(\mathbf{x}) := \mathbf{p}^T H(\mathbf{x})\mathbf{p},$$

it holds $g(\mathbf{x}^*) < 0$. Then, as $f \in C^2(\Omega)$ and so g is continuous in Ω, it exists $T \in \mathbb{R}, T > 0$ such that $\forall t \in [0, T]$

$$0 > g(\mathbf{x}^* + t\mathbf{p}) = \mathbf{p}^T H(\mathbf{x}^* + t\mathbf{p})\mathbf{p}. \tag{1.5}$$

From (1.2), $\forall \tau \in (0, T)$, it exists $t \in (0, 1)$ such that

$$f(\mathbf{x}^* + \tau\mathbf{p}) = f(\mathbf{x}^*) + \underbrace{\nabla f(\mathbf{x}^*)^T}_{=0 \text{ from Theorem 1.1.1}} \tau\mathbf{p} + \frac{1}{2}(\tau\mathbf{p})^T H(\mathbf{x}^* + \underbrace{t\tau}_{=t' \in (0,T)} \mathbf{p})\tau\mathbf{p} =$$

$$= f(\mathbf{x}^*) + \frac{1}{2}\tau^2 \underbrace{\mathbf{p}^T H(\mathbf{x}^* + t'\mathbf{p})\mathbf{p}}_{<0 \text{ from (1.5)}} < f(\mathbf{x}^*).$$

Then $\mathbf{x}^*$ cannot be a minimum point for f, which leads us to a contradiction. □

In the following theorem we show a sufficient condition: if this condition is satisfied, we are sure to have a minimum point. This is a second order condition: first order derivatives are not enough to establish a sufficient condition.

Theorem 1.1.3. *Sufficient second-order condition*
Let $f \in C^2(\Omega)$ for a neighbourhood Ω of $\mathbf{x}^*$.

$$\begin{cases} \nabla f(\mathbf{x}^*) = \mathbf{0} \\ H(\mathbf{x}^*) \text{ is positive definite} \end{cases} \implies \mathbf{x}^* \text{ is a minimum point for } f.$$

Proof. As $H(\mathbf{x}^*)$ is positive definite, it exists a neighbourhood $B = B_T(\mathbf{x}^*)$ of $\mathbf{x}^*$ such that $\forall \mathbf{x} \in B$ the matrix $H(\mathbf{x})$ remains positive definite. Then for every $\mathbf{p} \in \mathbb{R}^n$, $\forall \tau \in (0, T)$ it exists $t \in (0, 1)$ such that

$$\begin{aligned} f(\mathbf{x}^* + \tau\mathbf{p}) &= f(\mathbf{x}^*) + \underbrace{\nabla f(\mathbf{x}^*)^T}_{=\mathbf{0}} \tau\mathbf{p} + \frac{1}{2}(\tau\mathbf{p})^T H(\mathbf{x}^* + \underbrace{t\tau}_{=t' \in (0,T)} \mathbf{p})\tau\mathbf{p} = \\ &= f(\mathbf{x}^*) + \underbrace{\frac{1}{2}\tau^2 \, \mathbf{p}^T H(\underbrace{\mathbf{x}^* + t'\mathbf{p}}_{\in B})\mathbf{p}}_{>0} > f(\mathbf{x}^*), \end{aligned}$$

i.e. $f(\mathbf{x}^*)$ is the minimum value taken by f in B, so $\mathbf{x}^*$ is a minimum point for f. □

It is in general expensive to establish if $H(\mathbf{x}^*)$ is positive definite, as this requires the computation of the eigenvalues of the matrix. This condition is then usually not employed.

1.2 Convex functions

Let us now focus on a special case: that of convex functions.

Lemma 1.2.1. *Minima of convex functions. If f is convex, then every local minimum point for f is a global minimum point.*

Proof. Let $\mathbf{x}$ be a local minimum point for f and let us proceed by contradiction. Assume that $\mathbf{x}$ is not a global minimum, i.e. that it exists $\mathbf{y} \in A$ such that $f(\mathbf{y}) < f(\mathbf{x})$. Then, $\forall t \in [0, 1)$

$$f(t\mathbf{x} + (1-t)\mathbf{y}) \leq tf(\mathbf{x}) + (1-t)f(\mathbf{y}) < tf(\mathbf{x}) + (1-t)f(\mathbf{x}) = f(\mathbf{x}).$$

Then f has lower values in the points of the segment that connects $\mathbf{x}$ with $\mathbf{y}$ (except the point $\mathbf{x}$) than in $\mathbf{x}$, so $\mathbf{x}$ cannot be a local minimum point for f. □

Lemma 1.2.2. *Minima of strictly convex functions. If f is strictly convex it has just one minimum point.*

Proof. Let $\mathbf{x}$ be a minimum point for f and assume, by contradiction, that it exists another minimum point $\mathbf{y}$. From Lemma 1.2.1, all the minima points of f are global minima, so $f(\mathbf{x}) = f(\mathbf{y})$. Then $\forall t \in (0, 1)$

$$f(t\mathbf{x} + (1-t)\mathbf{y}) < tf(\mathbf{x}) + (1-t)f(\mathbf{y}) = tf(\mathbf{x}) + (1-t)f(\mathbf{x}) = f(\mathbf{x}).$$

Then f has lower values in the points of the segment that connects $\mathbf{x}$ with $\mathbf{y}$ (except for the endpoints) than in $\mathbf{x}$. Thus $\mathbf{x}$ cannot be a minimum point for f. □

1.3 Quadratic functions

A quadratic function is a function of the form

$$\begin{array}{rccl} q: & \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & \mathbf{x} & \longmapsto & q(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x} = \frac{1}{2}\sum_{i,j=1}^{n} x_i a_{ij} x_j - \sum_{i=1}^{n} b_i x_i, \end{array}$$

with $A \in \mathbf{R}^{n\times n}$ symmetric, $\mathbf{b} \in \mathbb{R}^n$.

Remark 1.3.1. *We can easily compute the gradient and Hessian for a quadratic function:*

- $\nabla q(\mathbf{x}) = A\mathbf{x} - \mathbf{b}$.
- $H(\mathbf{x}) = A$.

Definition 1.3.1. *$q(\mathbf{x})$ is positive definite if A is positive definite.*

Theorem 1.3.1. *A quadratic function that is positive definite is a strictly convex function.*

Theorem 1.3.2. *A positive definite quadratic function $q(\mathbf{x})$ has a unique minimizer $\mathbf{x}^*$, that is the unique solution of the problem*

$$A\mathbf{x} = \mathbf{b}.$$

Proof. As $q(\mathbf{x})$ is strictly convex, $q(\mathbf{x})$ has at most one minimizer. The Hessain matrix of $q(\mathbf{x})$ is A and it is positive definite, so from Theorem 1.1.3 and Theorem 1.1.1 it holds

$$\mathbf{x}^* \text{ is a minimizer of } q(\mathbf{x}) \quad \Longleftrightarrow \quad \mathbf{x}^* \text{ is a solution of } \nabla q(\mathbf{x}) = \mathbf{0},$$

i.e.

$$\mathbf{x}^* \text{ is a minimizer of } q(\mathbf{x}) \quad \Longleftrightarrow \quad \mathbf{x}^* \text{ is a solution of } A\mathbf{x} = \mathbf{b}.$$

A is positive definite and therefore invertible, so the problem $A\mathbf{x} = \mathbf{b}$ has a unique solution. □

Chapter 2

Iterative methods

In unconstrained optimization *iterative algorithms* are usually used to find a minimizer of f, that is algorithms such that, starting from an initial guess $\mathbf{x}_0 \in A$, builds a sequence $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ of points of A converging to a stationary point $\mathbf{x}^* \in A$ that satisfies the *simple decrease* property:

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$$

(or, as it happens in *nonmonotone* algorithms, that satisfies a condition such as $f(\mathbf{x}_{k+m}) < f(\mathbf{x}_k)$ for $m \geq 2$ fix). In this way $\mathbf{x}^*$ will surely not be a maximum point, but in unfortunate cases it may be a saddle point, there is usually no guarantee of convergence to a minimum, as the sufficient condition is usually not checked.

Usually convergence is proved in the limit for k that goes to infinity:

$$\lim_{k\to\infty} \mathbf{x_k} = \mathbf{x}^*,$$

but in practice the algorithm is stopped when a suitable stopping criterion is met. The stopping criterion is usually based on the norm of the gradient. As we want to reach a stationary point, we will have

$$\lim_{k\to\infty} \|\nabla f(\mathbf{x_k})\| = 0, \tag{2.1}$$

so, given a positive tolerance $\epsilon > 0$, the method is stopped as soon as $\|\nabla f(\mathbf{x_k})\| < \epsilon$. The magnitude of the tolerance depends on the specific application.

An algorithm is said to be *globally convergent*, if (2.1) is guaranteed for any initial guess $\mathbf{x}_0$, independently of the proximity of $\mathbf{x}_0$ to $\mathbf{x}^*$. Methods that are convergent just for initial guesses close enough to a minimum are said *locally convergent*. For such methods to be effective we need to have an a-priori information on the minimizers, which is not always available.

However, local methods can be made globally convergent by two different types of strategies: line-search and trust-region. These two strategies establish how to move from the current point $\mathbf{x}_k$ to the next one $\mathbf{x}_{k+1}$.

Line-search strategy In *line-search methods* we need to choose a descent direction $\mathbf{p}_k$. Then, the iterative scheme is as follows:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k,$$

a step is taken in the selected direction from the current point $\mathbf{x}_k$ whose length is $\alpha_k > 0$. We call $\alpha_k \mathbf{p}_k$ the step, $\mathbf{p}_k$ the step direction and α_k the step length. α_k is chosen in a way that

the following decrease condition is satisfied $f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) < f(\mathbf{x}_k)$. Ideally, one should choose $\alpha_k > 0$ that minimizes $\varphi(\alpha) = f(\mathbf{x}_k + \alpha \mathbf{p}_k)$, but to do that a minimization problem in $\mathbb{R}$ has to be solved at each iteration (find the points α such that $\varphi'(\alpha) = 0$). This would make the algorithm too expensive, except in some exceptional cases, that we will see. Different strategies are then used to make this choice.

Trust-region strategy The *trust-region strategy* is based on a quadratic model $m_k(\mathbf{x})$ that approximates $f(\mathbf{x})$ in a neighbourhood of the current position $\mathbf{x}_k$:

$$m_k(\mathbf{x}_k, \mathbf{p}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2} \mathbf{p}^T B_k \mathbf{p}, \qquad \mathbf{p} \in \mathbb{R}^n,$$

where B_k is $H(\mathbf{x}_k)$ (the Hessian matrix of f in $\mathbf{x}_k$) or an approximation to it.
We look for a step $\mathbf{p}_k$ that minimizes $m_k(\mathbf{x}_k, \mathbf{p})$ under the constraint that $\mathbf{x}_k + \mathbf{p}$ lays in a neighbourhood $B_{\Delta_k}(\mathbf{x}_k)$ of $\mathbf{x}_k$, called trust region, as it is the region in which we trust the model to be a good approximation to the function. [1].

We look for a step $\mathbf{p}_k$ solution of the problem

$$\min_{\mathbf{p} \in \mathbb{R}^n : \|\mathbf{p}\| \leq \Delta_k} m_k(\mathbf{x}_k, \mathbf{p}), \tag{2.2}$$

where $\Delta_k > 0$ is called trust-region radius.
As $p = 0$ belongs to the trust region, it holds $m_k(\mathbf{x}_k, \mathbf{p}_k) \leq m_k(\mathbf{x}_k, 0)$, but it may not hold $f(\mathbf{x}_k + \mathbf{p}_k) < f(\mathbf{x}_k)$. If this happens it means that the model is not a good approximation of $f(\mathbf{x})$ in the current trust region, the trust region is too large: we then chose $\Delta_{k+1} < \Delta_k$ and we solve again problem (2.2).
Otherwise we accept the step, i.e. we set

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k.$$

It is possible to show that after a finite number of steps $f(\mathbf{x}_k + \mathbf{p}_k) < f(\mathbf{x}_k)$.

The two strategies are based on different ideas; in line-search strategy we first choose the step direction and then we determine its length, in trust-region strategies first we choose the maximal length of the step (the trust-region radius) and then we determine the direction.
In the following we will use just the line-search strategy.

2.1 Directions for line-search methods

2.1.1 Direction of steepest descent

The steepest descent direction for f in $\mathbf{x}_k$ is

$$\mathbf{p}_k = -\nabla f(\mathbf{x}_k).$$

The direction $\mathbf{p}_k = -\nabla f(\mathbf{x}_k)$ is called of steepest descent for f in $\mathbf{x}_k$ because it is the direction in which, starting from $\mathbf{x}_k$, the values of f decrease the fastest. Indeed the direction of steepest

[1] Most often a ball is used as trust region, but other choices are possible depending on the problem, for example elliptic or rectangular trust regions (used for example when box constraints are present)

descent is the one that minimizes the directional derivative of f in $\mathbf{x}_k$:

$$\frac{\partial f}{\partial \mathbf{p}}(\mathbf{x}_k) = \nabla f(\mathbf{x}_k)^T \mathbf{p} \underbrace{=}_{\substack{\vartheta \in [0,\pi] \text{ is the angle} \\ \text{between } \nabla f(\mathbf{x}_k) \text{ and } \mathbf{p}}} \|\nabla f(\mathbf{x}_k)\| \|\mathbf{p}\| \cos\vartheta,$$

If we assume, without loss of generality, that $\|\mathbf{p}\| = 1$, the direction that maximises the decrease is the one that minimizes $\cos\vartheta$, so it must be such that $\vartheta = \pi$ and so

$$p_k = \frac{-\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|}.$$

An advantage of the steepest descent method (the one that uses this direction) is that it requires just the computation of the gradient of f at each iteration, and not that of the second order derivatives. However the convergence is generally really slow (it requires a large number of iterations to reach a stationary point).

2.1.2 Newton's direction

Let us consider the quadratic model of f in $\mathbf{x}_k$:

$$m_k(\mathbf{p}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T H(\mathbf{x}_k)\mathbf{p},$$

and let assume $H(\mathbf{x}_k)$ to be positive definite.
Newton's direction $\mathbf{p}_k^N$ is the minimizer of $m_k(\mathbf{p})$, i.e., being $m_k(\mathbf{p})$ a positive definite quadratic function, it is the solution of Newton's system

$$H(\mathbf{x}_k)\mathbf{p} = -\nabla f(\mathbf{x}_k). \tag{2.3}$$

The analytic expression of Newton's direction is then

$$\mathbf{p}_k^N = -H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)$$

(note however that in practice $H(\mathbf{x}_k)$ is never explicitly inverted to compute such a direction, Newton's system is rather solved.)

Thanks to the fact that $H(\mathbf{x}_k)$ is positive definite, we have not only that $H(\mathbf{x}_k)$ is invertible, so Newton's system has one and just one solution, but it also holds that $\mathbf{p}_k^N$ is a descent direction, as

$$\nabla f(\mathbf{x}_k)^T \mathbf{p}_k^N = \nabla f(\mathbf{x}_k)^T \Big(- H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)\Big) = -\nabla f(\mathbf{x}_k)^T H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k) < 0,$$

because $H(\mathbf{x}_k)^{-1}$ is positive definite.
If $H(\mathbf{x}_k)$ is not positive definite, not only $\mathbf{p}_k^N$ may not be well-defined ($H(\mathbf{x}_k)$ may not be invertible and so (2.3) may not have a unique solution), but even if $\mathbf{p}_k^N$ is well-defined, $\mathbf{p}_k^N$ may not be a descent direction.
Methods based on Newton's direction are usually characterized by fast local convergence (they require few iterations to converge), but they are expensive as they require not only the computation of $H(\mathbf{x}_k)$ at each step, but also the solution of the linear system (2.3) that may be expensive if the size of the problem is large.

2.1.3 Quasi-Newton directions

Let us consider a quadratic model for f:

$$m_k(\mathbf{p}) = f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T B_k \mathbf{p},$$

where $B_k \approx H(\mathbf{x}_k)$ is a SPD (symmetric positive definite) matrix.
Quasi-Newton direction $\mathbf{p}_k$ is the minimizer of $m_k(\mathbf{p})$, i.e. is the solution of quasi-Newton system

$$B_k \mathbf{p} = -\nabla f(\mathbf{x}_k);$$

the analytical expression of the quasi-Newton direction is

$$\mathbf{p}_k^{QN} = -B_k^{-1} \nabla f(\mathbf{x}_k).$$

2.2 Rates of convergence

Let $\{\mathbf{x}_k\}_{k\in\mathbb{N}}$ be a sequence of elements of $\mathbb{R}^n$ converging to $\mathbf{x}^*$. The speed at which a convergent sequence approaches its limit is represented by its order of convergence and by its rate of convergence. The sequence is said to have order of convergence $q \geq 1$ and rate of convergence μ if

$$\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|^q} = \mu.$$

- The sequence is said to converge *linearly* if it exists $r \in (0,1)$ such that

$$\lim_{k\to+\infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = r.$$

- The sequence is said to converge *superlinearly* (faster than linearly) if

$$\lim_{k\to+\infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = 0.$$

- The sequence is said to converge *sublinearly* (slower than linearly) if

$$\lim_{k\to+\infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} = 1.$$

- The convergence is said to be *quadratic* if it exists $M > 0$ such that

$$\lim_{k\to+\infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^2} < M.$$

- In general, given $p > 1$, the sequence is said to converge with order p if it exists $M > 0$ such that

$$\lim_{k\to+\infty} \frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|^p} < M.$$

  In particular, convergence with order

- $p = 1$ is called linear convergence,
- $p = 2$ is called quadratic convergence,
- $p = 3$ is called cubic convergence.

If the convergence is linear it means that

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq r\|\mathbf{x}_k - \mathbf{x}^*\|$$

for all k large enough. This means that (asymptotically) the distance of the solution approximation from the solution at step $k+1$ ($\|\mathbf{x}_{k+1} - \mathbf{x}^*\|$) is lower than a fraction of the distance of the solution approximation from the solution at the previous step ($r\|\mathbf{x}_k - \mathbf{x}^*\|$): at each iteration this distance is decreased and the rate at which it is decreased depends on r, if r is close to one the decrease is really slow.

Usually the cost of a method is directly proportional to the speed of convergence: generally an expensive method (for which a single iteration is expensive to compute) has a higher rate of convergence and requires less iterations to converge. For example methods based on Newton's directions enjoy a quadratic local rate of convergence. Quasi-Newton methods are less expensive then Newton's method, but this is paid with a slower superlinear convergence.

2.3 Steepest descent method for quadratic functions

We have seen that when using line-search strategies it is in general too expensive to choose α_k solving the minimization problem

$$\min_{\alpha>0} \varphi(\alpha)$$

with $\varphi(\alpha) = f(x_k + \alpha p_k)$, given the current iterate x_k and a descent direction p_k. In particular cases the solution of this minimization problem can be computed analytically, and so the optimal value can be employed in a cheap way. This is the case when the steepest descent direction is used for quadratic functions.

Let us consider the positive definite quadratic function

$$\begin{array}{rccl} q: & \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & \mathbf{x} & \longmapsto & q(\mathbf{x}) = \dfrac{1}{2}\mathbf{x}^T A\mathbf{x} - \mathbf{b}^T\mathbf{x}. \end{array}$$

We know that $\mathbf{x}^*$ is a minimizer for $q(\mathbf{x})$ if and only if $\nabla q(\mathbf{x}^*) = \mathbf{0}$.
We choose the steepest descent direction

$$\mathbf{p}_k = -\nabla q(\mathbf{x}_k) = -(A\mathbf{x}_k - \mathbf{b}) := -\mathbf{g}_k,$$

and we use the line-search method

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k = \mathbf{x}_k - \alpha_k \mathbf{g}_k.$$

In this particular case it is possible to choose $\alpha_k \in \mathbb{R}$ that exactly minimizes

$$\varphi(\alpha) \quad = \quad q(\mathbf{x}_k + \alpha\mathbf{p}_k) \quad = \quad q(\mathbf{x}_k - \alpha\mathbf{g}_k) \quad = \quad \frac{1}{2}(\mathbf{x}_k - \alpha\mathbf{g}_k)^T A(\mathbf{x}_k - \alpha\mathbf{g}_k) - (\mathbf{x}_k - \alpha\mathbf{g}_k)^T\mathbf{b}.$$

The minimizer can indeed be analytically computed. Performing the computations and by remarking that

$$\underbrace{\mathbf{x}_k^T A\mathbf{g}_k}_{\in\mathbb{R}} = (\mathbf{x}_k^T A\mathbf{g}_k)^T = \mathbf{g}_k^T A^T\mathbf{x}_k \underbrace{=}_{A \text{ is symmetric}} \mathbf{g}_k^T A\mathbf{x}_k,$$

we obtain that

$$\begin{aligned}\varphi(\alpha) &= \frac{1}{2}\mathbf{g}_k^T A\mathbf{g}_k\alpha^2 - \mathbf{x}_k^T A\mathbf{g}_k\alpha + \mathbf{b}^T\mathbf{g_k}\alpha + \frac{1}{2}\mathbf{x}_k^T A\mathbf{x}_k - \mathbf{x}_k^T\mathbf{b} = \\ &= \frac{1}{2}\mathbf{g}_k^T A\mathbf{g}_k\alpha^2 - \mathbf{g}_k^T\mathbf{g}_k\alpha + \frac{1}{2}\mathbf{x}_k^T A\mathbf{x}_k - \mathbf{x}_k^T\mathbf{b},\end{aligned}$$

i.e., $\varphi(\alpha)$ is a parabola with branches pointing up, since $\frac{1}{2}\mathbf{g}_k^T A\mathbf{g}_k > 0$, as A is positive definite. Then the minimizer of $\varphi(\alpha)$ is α such that

$$0 = \varphi'(\alpha) = \mathbf{g}_k^T A\mathbf{g}_k\alpha - \mathbf{g}_k^T\mathbf{g}_k,$$

that is

$$\alpha_k = \frac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T A\mathbf{g}_k} = \frac{\|\nabla\mathbf{q}(\mathbf{x}_k)\|^2}{\|\nabla\mathbf{q}(\mathbf{x}_k)\|_A^2},$$

having defined $\forall\mathbf{y} \in \mathbb{R}^n$, $\forall A \in \mathbb{R}^{n\times n}$ SPD the energy norm $\|\mathbf{y}\|_A^2 = \mathbf{y}^T A\mathbf{y}$. Remark that $\alpha_k > 0$: that means that we will go along the direction $\mathbf{p}_k$, and not in the opposite one.

We derive then the following algorithm for the steepest descent method or gradient method.

Algorithm for gradient method (first version)

0. Given $\mathbf{x}_0, A, \mathbf{b}, \text{toll}$
1. Compute $\mathbf{g}_0 = A\mathbf{x}_0 - \mathbf{b}$
2. For $k = 0, 1, \ldots$
   1. Compute $\alpha_k = \dfrac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T A\mathbf{g}_k}$
   2. Set $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k\mathbf{g}_k$
   3. Compute $\mathbf{g}_{k+1} = A\mathbf{x}_{k+1} - \mathbf{b}$
   4. If $\|\mathbf{g}_{k+1}\| \leq \text{toll}$ return $\mathbf{x}_{k+1}$ and stop.

At each iteration two matrix-vector products are performed: $A\mathbf{g}_k$ and $A\mathbf{x}_{k+1}$. The algorithm can be improved to require just one matrix vector product at each iteration, thanks to the fact that

$$\mathbf{g}_{k+1} = A\mathbf{x}_{k+1} - \mathbf{b} = A(\mathbf{x}_k - \alpha_k\mathbf{g}_k) - \mathbf{b} = A\mathbf{x}_k - \alpha_k A\mathbf{g}_k - \mathbf{b} = \mathbf{g}_k - \alpha_k A\mathbf{g}_k.$$

We derive then the following optimized version of the algorithm.

Algorithm for gradient method (optimized version)

0. Given $\mathbf{x}_0, A, \mathbf{b}, \text{toll}$
1. Compute $\mathbf{g}_0 = A\mathbf{x}_0 - \mathbf{b}$
2. For $k = 0, 1, \ldots$
   1. Compute $\mathbf{r}_k = A\mathbf{g}_k$
   2. Compute $\alpha_k = \dfrac{\mathbf{g}_k^T\mathbf{g}_k}{\mathbf{g}_k^T\mathbf{r}_k}$

3. Set $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k \mathbf{g}_k$
4. Compute $\mathbf{g}_{k+1} = \mathbf{g}_k - \alpha_k \mathbf{r}_k$
5. If $\|\mathbf{g}_{k+1}\| \leq$ toll return $\mathbf{x}_{k+1}$ and stop

This algorithm has then a really low per-iteration cost. The memory consumption is also low: at each iteration it requires to memorize the vector $\mathbf{g}_k$, and no matrices.
The convergence is on the contrary slow: we can prove that it holds

$$\frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|_A}{\|\mathbf{x}_k - \mathbf{x}^*\|_A} \leq \frac{k_2(A) - 1}{k_2(A) + 1},$$

where $k_2(A)$ is the condition number of A in the 2-norm. [2] The convergence of the method is then linear.
If a method converges linearly it exists $r \in (0, 1)$ such that

$$\frac{\|\mathbf{x}_{k+1} - \mathbf{x}^*\|}{\|\mathbf{x}_k - \mathbf{x}^*\|} \leq r$$

for each k sufficiently large. The more the constant r is close to 0 the faster the method converges.
In this case, the closer $\dfrac{k_2(A) - 1}{k_2(A) + 1}$ is to 0, the faster the method converges, i.e., the closer $k_2(A)$ is to 1, i.e. if A is well-conditioned.
Methods with linear convergence are in general not well suited for problems in which a high accuracy (low toll) is required, because they will need a large number of iterations to find the desired solution approximation.
Figure 2.3 shows possible sequences of iterations generated by the steepest descent method (or gradient method) applied to an elliptic quadratic function $q(\mathbf{x}) = q(x_1, x_2)$ (a quadratic function that has ellipses as level curves). The convergence depends on the choice of the starting guess and of the step length. This is particularly evident when the ellipses have the two centers that are far from each other. If on the contrary the level curves are circles, the minimizer $\mathbf{x}^*$ is easily obtained in just one iteration: $\mathbf{x}_1 = \mathbf{x}^*$.

2.4 Convergence of Newton's method

What is usually called the "pure Newton's method" is based on the following iterative scheme:

$$\left\{ \begin{array}{c} \text{Given } \mathbf{x}_0 \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k^N \end{array} \right\}$$

[2] $k_2(A) = \|A\|_2 \|A^{-1}\|_2$. Being A SPD, if $\sigma(A) = \{\lambda_1, \ldots, \lambda_n\}$ is the spectre of A with $0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n$, it holds $k_2(A) = \dfrac{\lambda_n}{\lambda_1}$. Indeed

$$\|A\|_2 \underbrace{=}_{\text{definition}} \sqrt{\rho(A^T A)} \underbrace{=}_{A \text{ is symmetric}} \sqrt{\rho(A^2)} = \sqrt{\rho(A)^2} = \sqrt{\lambda_n^2} \underbrace{=}_{\lambda_n > 0} \lambda_n.$$

The 2-norm of a SPD matrix is equal to its largest eigenvalue, so $\|A^{-1}\|_2 = \dfrac{1}{\lambda_1}$, and then

$$k_2(A) = \|A\|_2 \|A^{-1}\|_2 = \frac{\lambda_n}{\lambda_1}.$$

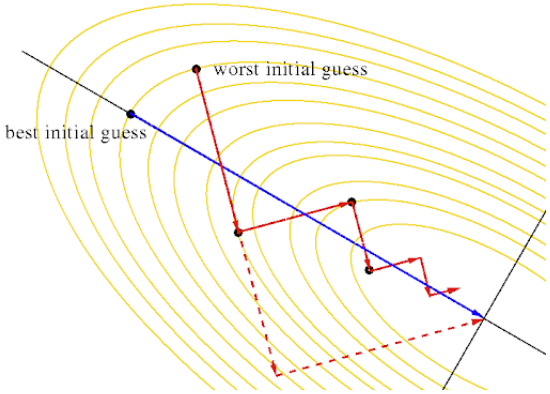

Figure 2.1: Sequence of iterations generated by the steepest descent method applied to a quadratic function. The convergence depends on the choice of the starting guess and of the step length.

As shown in the following theorem it is a locally convergent method, and the local rate of convergence is quadratic. This is not a globally convergent method and it can be made so by coupling it with a line-search strategy, as we will see in the next chapter.

Theorem 2.4.1. *Local convergence of Newton's method. Let $\mathbf{x}^*$ be a minimizer for f, Ω be a neighbourhood of $\mathbf{x}^*$, $f \in C^2(\Omega)$, $H(\mathbf{x}^*)$ positive definite and $H(\mathbf{x})$ Lipschitz continuous in Ω with Lipschitz constant L.*
It exists $\rho > 0$ such that if $\mathbf{x}_0 \in B_\rho(\mathbf{x}^*)$ then the sequence $\{\mathbf{x}_k\}$ built by Newton's method is well-defined [3], converges to $\mathbf{x}^*$ quadratically and $\|\nabla f(\mathbf{x}_k)\|$ converges to 0 quadratically.

Proof. By assumption it exists $r > 0$ such that $\forall \mathbf{x} \in B_r(\mathbf{x}^*)$ $H(\mathbf{x})$ is positive definite (so invertible) and it holds (see TD2)

$$\|H(\mathbf{x})^{-1}\| \leq 2\|H(\mathbf{x}^*)^{-1}\|. \tag{2.4}$$

We can assume $B_r(\mathbf{x}^*) \subseteq \Omega$.
If $\mathbf{x}_k \in B_r(\mathbf{x}^*)$ then

$$\mathbf{x}_{k+1} - \mathbf{x}^* = \mathbf{x}_k + \mathbf{p}_k^N - \mathbf{x}^* = \mathbf{x}_k - \mathbf{x}^* - H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k) = H(\mathbf{x}_k)^{-1}\Big(H(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{x}^*) - \nabla f(\mathbf{x}_k)\Big)$$
$$= H(\mathbf{x}_k)^{-1}\Big(H(\mathbf{x}_k)(\mathbf{x}_k - \mathbf{x}^*) + \overbrace{\nabla f(\mathbf{x}^*)}^{=\mathbf{0}} - \nabla f(\mathbf{x}_k)\Big).$$

Because

$$\nabla f(\mathbf{x}^*) - \nabla f(\mathbf{x}_k) = \Big[\nabla f(\mathbf{x}_k + t(\mathbf{x}^* - \mathbf{x}_k))\Big]_{t=0}^{t=1} = \int_0^1 H(\mathbf{x}_k + t(\mathbf{x}^* - \mathbf{x}_k))(\mathbf{x}^* - \mathbf{x}_k)\, \mathrm{d}t,$$

[3] We can build it because $H(\mathbf{x}_k)$ is positive definite for each $k \in \mathbb{N}$.

we have

$$\begin{aligned}\mathbf{x}_{k+1}-\mathbf{x}^* &= H(\mathbf{x}_k)^{-1}\Big(H(\mathbf{x}_k)(\mathbf{x}_k-\mathbf{x}^*)+\int_0^1 H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))(\mathbf{x}^*-\mathbf{x}_k)\,\mathrm{d}t\Big)\\ &= H(\mathbf{x}_k)^{-1}\Big(\underbrace{H(\mathbf{x}_k)(\mathbf{x}_k-\mathbf{x}^*)}_{\text{does not depend on } t}-\int_0^1 H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))(\mathbf{x}_k-\mathbf{x}^*)\,\mathrm{d}t\Big)\\ &= H(\mathbf{x}_k)^{-1}\int_0^1\Big(H(\mathbf{x}_k)-H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big)(\mathbf{x}_k-\mathbf{x}^*)\,\mathrm{d}t.\end{aligned}$$

Passing to norms we obtain that

$$\begin{aligned}\|\mathbf{x}_{k+1}-\mathbf{x}^*\| &\le \|H(\mathbf{x}_k)^{-1}\|\left\|\int_0^1\Big(H(\mathbf{x}_k)-H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big)(\mathbf{x}_k-\mathbf{x}^*)\,\mathrm{d}t\right\| \underbrace{\le}_{(2.4)}\\ &\le 2\|H(\mathbf{x}^*)^{-1}\|\left\|\int_0^1\Big(H(\mathbf{x}_k)-H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big)(\mathbf{x}_k-\mathbf{x}^*)\,\mathrm{d}t\right\|\\ &\le 2\|H(\mathbf{x}^*)^{-1}\|\int_0^1\Big\|H(\mathbf{x}_k)-H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big\|\|\mathbf{x}_k-\mathbf{x}^*\|\,\mathrm{d}t\\ &= 2\|H(\mathbf{x}^*)^{-1}\|\|\mathbf{x}_k-\mathbf{x}^*\|\int_0^1\Big\|H(\mathbf{x}_k)-H(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big\|\,\mathrm{d}t\\ &\le 2\|H(\mathbf{x}^*)^{-1}\|\|\mathbf{x}_k-\mathbf{x}^*\|\int_0^1 L\Big\|\mathbf{x}_k-(\mathbf{x}_k+t(\mathbf{x}^*-\mathbf{x}_k))\Big\|\,\mathrm{d}t\\ &= 2\underbrace{L\|H(\mathbf{x}^*)^{-1}\|}_{\text{we call this }\widetilde{L}}\|\mathbf{x}_k-\mathbf{x}^*\|\int_0^1\|-t(\mathbf{x}^*-\mathbf{x}_k)\|\,\mathrm{d}t\\ &= 2\widetilde{L}\|\mathbf{x}_k-\mathbf{x}^*\|^2\underbrace{\int_0^1 t\,\mathrm{d}t}_{=\left[\frac{t^2}{2}\right]_0^1=\frac{1}{2}} = \widetilde{L}\|\mathbf{x}_k-\mathbf{x}^*\|^2.\end{aligned}$$

We have proved that if $\mathbf{x}_k\in B_r(\mathbf{x}^*)$ then

$$\|\mathbf{x}_{k+1}-\mathbf{x}^*\|\le\widetilde{L}\|\mathbf{x}_k-\mathbf{x}^*\|^2. \tag{2.5}$$

Let $\rho=\min\left\{r,\dfrac{1}{2\widetilde{L}}\right\}$.
Let us assume that $\mathbf{x}_0\in B_\rho(\mathbf{x}^*)$. Because $\mathbf{x}_0\in B_\rho(\mathbf{x}^*)\underbrace{\subseteq}_{\rho\le r}B_r(\mathbf{x}^*)$, it follows

$$\|\mathbf{x}_1-\mathbf{x}^*\| \le \widetilde{L}\|\mathbf{x}_0-\mathbf{x}^*\|^2 \le \widetilde{L}\rho^2 \underbrace{\le}_{\rho\le\frac{1}{2\widetilde{L}}\implies\widetilde{L}\rho\le\frac{1}{2}} \frac{1}{2}\rho,$$

i.e., $\mathbf{x}_1\in B_\rho(\mathbf{x}^*)$. Because $\mathbf{x}_1\in B_\rho(\mathbf{x}^*)\underbrace{\subseteq}_{\rho\le r}B_r(\mathbf{x}^*)$, analogously it holds $\|\mathbf{x}_2-\mathbf{x}^*\|<\dfrac{1}{2}\rho$, i.e., $\mathbf{x}_2\in B_\rho(\mathbf{x}^*)$, and so on. Then by induction we have that

$$\mathbf{x}_k\in B_\rho(\mathbf{x}^*)\qquad\forall k\in\mathbb{N}.$$

This ensures that the sequence $\{\mathbf{x}_k\}$ built by Newton's method is well-defined because the matrices $H(\mathbf{x}_k)$ are all positive definite from (2.4). Moreover, from (2.5)

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \widetilde{L}\|\mathbf{x}_k - \mathbf{x}^*\|^2 \qquad \forall k \in \mathbb{N}. \tag{2.6}$$

Then $\forall k \in \mathbb{N}$

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \quad \leq \quad \widetilde{L}\|\mathbf{x}_k - \mathbf{x}^*\|^2 \quad \leq \quad \widetilde{L}\rho\|\mathbf{x}_k - \mathbf{x}^*\| \quad \underbrace{\leq}_{\rho \leq \frac{1}{2\widetilde{L}} \Longrightarrow \widetilde{L}\rho \leq \frac{1}{2}} \quad \frac{1}{2}\|\mathbf{x}_k - \mathbf{x}^*\|,$$

i.e.

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \frac{1}{2}\|\mathbf{x}_k - \mathbf{x}^*\| \qquad \forall k \in \mathbb{N}$$

and so

$$\|\mathbf{x}_k - \mathbf{x}^*\| \leq \frac{1}{2}\|\mathbf{x}_{k-1} - \mathbf{x}^*\| \leq \left(\frac{1}{2}\right)^2 \|\mathbf{x}_{k-2} - \mathbf{x}^*\| \leq \ldots < \left(\frac{1}{2}\right)^k \|\mathbf{x}_0 - \mathbf{x}^*\| \quad \xrightarrow[k\to+\infty]{} \quad 0,$$

The sequence $\{\mathbf{x}_k\}$ converges then to $\mathbf{x}^*$ and from (2.6) we have that the convergence is quadratic.
As x^* is a stationary point by assumption, the sequence $\{\|\nabla f(\mathbf{x}_k)\|\}$ converges to 0 by continuity. We remark that

$$\|\nabla f(\mathbf{x}_{k+1})\| = \|\nabla f(\mathbf{x}_{k+1}) - \overbrace{(H(\mathbf{x}_k)\mathbf{p}_k^N + \nabla f(\mathbf{x}_k))}^{=\mathbf{0}}\| = \|\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) - H(\mathbf{x}_k)\mathbf{p}_k^N\|.$$

Because

$$\nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) = \Big[\nabla f(\mathbf{x}_k + t(\mathbf{x}_{k+1} - \mathbf{x}_k))\Big]_{t=0}^{t=1} = \Big[\nabla f(\mathbf{x}_k + t\mathbf{p}_k^N)\Big]_{t=0}^{t=1} = \int_0^1 H(\mathbf{x}_k + t\mathbf{p}_k^N)\mathbf{p}_k^N \, \mathrm{d}t,$$

we have

$$\begin{aligned}
\|\nabla f(\mathbf{x}_{k+1})\| &= \left\| \int_0^1 H(\mathbf{x}_k + t\mathbf{p}_k^N)\mathbf{p}_k^N \, \mathrm{d}t - \underbrace{H(\mathbf{x}_k)\mathbf{p}_k^N}_{\text{does not depend on } t} \right\| = \left\| \int_0^1 \Big(H(\mathbf{x}_k + t\mathbf{p}_k^N) - H(\mathbf{x}_k)\Big)\mathbf{p}_k^N \, \mathrm{d}t \right\| \\
&\leq \int_0^1 \|H(\mathbf{x}_k + t\mathbf{p}_k^N) - H(\mathbf{x}_k)\|\|\mathbf{p}_k^N\| \, \mathrm{d}t \quad = \quad \|\mathbf{p}_k^N\| \int_0^1 \|H(\mathbf{x}_k + t\mathbf{p}_k^N) - H(\mathbf{x}_k)\| \, \mathrm{d}t \\
&\leq \quad \|\mathbf{p}_k^N\| \int_0^1 L\|\mathbf{x}_k + t\mathbf{p}_k^N - \mathbf{x}_k\| \, \mathrm{d}t \quad = \quad L\|\mathbf{p}_k^N\| \int_0^1 \|t\mathbf{p}_k^N\| \, \mathrm{d}t \\
&= L\|\mathbf{p}_k^N\|^2 \underbrace{\int_0^1 t \, \mathrm{d}t}_{=\left[\frac{t^2}{2}\right]_0^1 = \frac{1}{2}} \quad = \quad \frac{1}{2}L\|\mathbf{p}_k^N\|^2 \quad = \quad \frac{1}{2}L\|H(\mathbf{x}_k)^{-1}\nabla f(\mathbf{x}_k)\|^2 \\
&\leq \quad \frac{1}{2}L\|H(\mathbf{x}_k)^{-1}\|^2\|\nabla f(\mathbf{x}_k)\|^2 \quad \underbrace{\leq}_{(2.4)} \quad \frac{1}{2}L4\|H(\mathbf{x}^*)^{-1}\|^2\|\nabla f(\mathbf{x}_k)\|^2 \\
&= \quad 2L\|H(\mathbf{x}^*)^{-1}\|^2\|\nabla f(\mathbf{x}_k)\|^2 := M\|\nabla f(\mathbf{x}_k)\|^2.
\end{aligned}$$

Because M does not depend on k, we have proved that it exists $M > 0$ such that $\forall k \in \mathbb{N}$

$$\big| \, \|\nabla f(\mathbf{x}_{k+1})\| - 0 \, \big| \leq M \big| \, \|\nabla f(\mathbf{x}_k)\| - 0 \, \big|^2,$$

i.e. the convergence of $\{\|\nabla f(\mathbf{x}_k)\|\}$ to 0 is quadratic. □

Chapter 3

Line-search methods

In this chapter we will introduce and analyse the line-search methods for nonlinear optimization problems.

The crucial operation in line-search methods is the computation of the step-length, for which we have to face a tradeoff. We would like to choose α_k to have a substantial reduction of f, but at the same time we do not want to spend too much time making this choice. In particular, we will see that the asking the simple decrease of f is not a sufficient condition to get a convergent method, and that we will require some other conditions on the step-length to avoid too small or too long steps. Let's see why these are needed by some examples.

3.1 Armijo and Wolfe conditions

Example 1 (Too long steps)

Let us consider $f(x) = x^2$, $x_0 = 2$, $p_k = (-1)^{k+1}$ (these are descent directions), $\alpha_k = 2 + \frac{3}{2^{k+1}}$. The sequence x_k built iteratively starting from x_0 setting $x_{k+1} = x_k + \alpha_k p_k$ is

$$x_k = (-1)^k \left(1 + \frac{1}{2^k}\right), \qquad k \in \mathbb{N}.$$

We can prove this by induction on k.

$$x_1 = x_0 + \alpha_0 p_0 = 2 + \left(2 + \frac{3}{2}\right)(-1) = -\frac{3}{2}.$$

If the thesis holds for k,

$$x_{k+1} = x_k + \alpha_k p_k = (-1)^k \left(1 + \frac{1}{2^k}\right) + \left(2 + \frac{3}{2^{k+1}}\right)(-1)^{k+1} =$$

$$= (-1)^{k+1} \left(-1 - \frac{1}{2^k} + 2 + \frac{3}{2^{k+1}}\right) = (-1)^{k+1} \left(1 - \frac{3-2}{2^{k+1}}\right) = (-1)^{k+1} \left(1 + \frac{1}{2^{k+1}}\right).$$

The simple decrease condition is satisfied because

$$f(x_{k+1}) = x_{k+1}^2 = \left(1 + \frac{1}{2^{k+1}}\right)^2 < \left(1 + \frac{1}{2^k}\right)^2 = x_k^2 = f(x_k),$$

but x_k does not converge to the minimizer of $f(x)$, $x^* = 0$, because

$$x_{2k} = 1 + \frac{1}{2^{2k}} \xrightarrow[k\to+\infty]{} 1, \qquad x_{2k+1} = -\left(1 + \frac{1}{2^{2k+1}}\right) \xrightarrow[k\to+\infty]{} -1.$$

The simple decrease condition is not sufficient to guarantee a convergence of $\{x_k\}$ to $x^* = 0$ because we have chosen a sequence of α_k with too large values.

Example 2 (Too short steps)

Let us consider $f(x) = x^2$, $x_0 = 2$, $p_k = -1$ (these are descent directions), $\alpha_k = \frac{1}{2^{k+1}}$. The sequence x_k built iteratively starting from x_0 setting $x_{k+1} = x_k + \alpha_k p_k$ is

$$x_k = 1 + \frac{1}{2^k}, \qquad k \in \mathbb{N}.$$

We can prove it by induction on k:

$$x_1 = x_0 + \alpha_0 p_0 = 2 + \frac{1}{2}(-1) = 1 + \frac{1}{2}.$$

If the thesis holds for k,

$$x_{k+1} = x_k + \alpha_k p_k = 1 + \frac{1}{2^k} + \frac{1}{2^{k+1}}(-1) = 1 + \frac{1}{2^k} - \frac{1}{2^{k+1}} = 1 + \frac{2-1}{2^{k+1}} = 1 + \frac{1}{2^{k+1}}.$$

The simple decrease condition is satisfied as

$$f(x_{k+1}) = x_{k+1}^2 = \left(1 + \frac{1}{2^{k+1}}\right)^2 < \left(1 + \frac{1}{2^k}\right)^2 = x_k^2 = f(x_k),$$

but x_k does not converge to $x^* = 0$ minimizer of $f(x)$ because

$$x_k = 1 + \frac{1}{2^k} \xrightarrow{k\to+\infty} 1.$$

The simple decrease condition is not sufficient to guarantee a convergence of $\{x_k\}$ to $x^* = 0$ because we have chosen a sequence of α_k with too small values.

We need then additional conditions.

Armijo rule

Armijo rule (A) requires that

$$f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \le f(\mathbf{x}_k) + \alpha_k c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k, \quad c_1 \in (0,1), \tag{A}$$

usually $c_1 = 10^{-4}$. (A) is stronger than just asking the simple decrease $f(\mathbf{x}_{k+1}) \le f(\mathbf{x}_k)$ because $\nabla f(\mathbf{x}_k)^T \mathbf{p}_k < 0$.

Let

$$\varphi(\alpha) = f(\mathbf{x}_k + \alpha \mathbf{p}_k),$$
$$\ell(\alpha) = f(\mathbf{x}_k) + \alpha c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k,$$

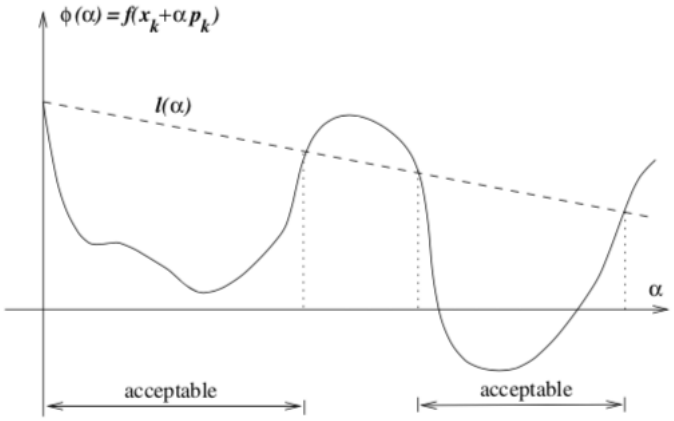

Figure 3.1: Parameters α that satisfy (A)

Armijo rule requires $\varphi(\alpha)$ to be below the line $\ell(\alpha)$, i.e., that $\varphi(\alpha) \leq \ell(\alpha)$.

The slope of $\varphi(\alpha)$ is $\varphi'(\alpha) = \nabla f(\mathbf{x}_k + \alpha \mathbf{p}_k)^T \mathbf{p}_k$, for $\alpha_k = 0$ this is $\nabla f(\mathbf{x}_k)^T \mathbf{p}_k < 0$. The slope of $\ell(\alpha)$ is $c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k = c_1 \varphi'(0) < 0$. Because $c_1 < 1$ and the two terms are negative, it follows

$$c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k > \nabla f(\mathbf{x}_k)^T \mathbf{p}_k,$$

that is the line $\ell(\alpha)$ lies above the graph of φ for small positive values of (α).

Choosing α_k according to (A) avoids choosing α_k too large, as in Example 1. However, this condition alone is not sufficient to ensure the algorithm to make reasonable progress, because too small steps may be taken. We then introduce also the following condition, to rule out unacceptably small steps.

Wolfe rule

Wolfe rule requires that

$$\nabla f(\mathbf{x}_k + \alpha_k \mathbf{p}_k)^T \mathbf{p}_k \geq c_2 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k \quad c_2 \in (c_1, 1). \tag{W}$$

The first term $\nabla f(\mathbf{x}_k + \alpha_k \mathbf{p}_k)^T \mathbf{p}_k = \varphi'(\alpha_k)$ is the slope of $\varphi(\alpha)$. The condition requires this slope to be greater than the negative slope $c_2 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k$ that in Figure 3.2 is labelled as the desired slope. If the slope of $\varphi'(\alpha_k)$ is strongly negative, it means that we can reduce f significantly by moving further along the chosen direction. On the other hand, if the slope is only slightly negative we cannot expect much more decrease in f in this direction and we can terminate the line-search. Usually $c_2 = 0.9$, for example when p_k is the Newton or quasi-Newton direction.

Because $c_1 < c_2 < 1$ and $\nabla f(\mathbf{x}_k)^T \mathbf{p}_k$ is negative it holds

$$c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k > c_2 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k > \nabla f(\mathbf{x}_k)^T \mathbf{p}_k,$$

i.e. the desired slope is between those of $\ell(\alpha)$ and $\varphi(\alpha)$.

Choosing α_k satisfying (W) avoids choosing α_k too small, as it happens in Example 2.

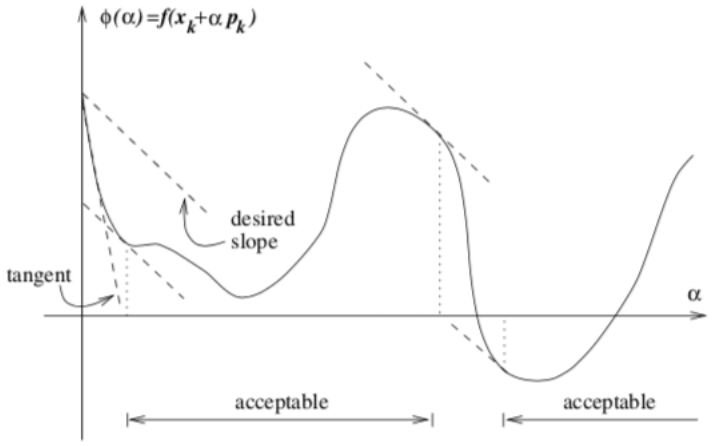

Figure 3.2: Parameters α that satisfy (W)

Lemma 3.1.1. *Wolfe's Lemma*

Let $f : \mathbb{R}^n \to \mathbb{R}$ continuously differentiable and bounded below in $\{\mathbf{x}_k + \alpha\mathbf{p}_k \mid \alpha > 0\}$, with $\mathbf{p}_k$ a descent direction for f in $\mathbf{x}_k$, and let $c_1, c_2 : 0 < c_1 < c_2 < 1$.
It exists $I \subseteq (0, +\infty)$ non empty such that every $\alpha \in I$ satisfies (A) + (W).

Proof. Let $g(\alpha) = \varphi(\alpha) - \ell(\alpha)$. (A) requires that

$$g(\alpha) \leq 0.$$

Because

$$g(0) = \varphi(0) - \ell(0) = f(\mathbf{x}_k) - f(\mathbf{x}_k) = 0$$

and

$$g'(0) = \varphi'(0) - \ell'(0) = \nabla f(\mathbf{x}_k)^T\mathbf{p}_k - c_1\nabla f(\mathbf{x}_k)^T\mathbf{p}_k = \underbrace{(1 - c_1)}_{>0}\underbrace{\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}_{<0} < 0,$$

it means $g(0) = 0$ and then decreases. As $g \in C^0$ because $f \in C^1$, it exists a right neighbourhood of 0 where $g(\alpha) < 0$. Let $\overline{\alpha}$ be the smallest positive zero of $g(\alpha)$.[1] It holds $g(\alpha) \leq 0$, $\forall\alpha \in [0, \overline{\alpha}]$, that is all the $\alpha \in [0, \overline{\alpha}]$ satisfy (A). In particular, in $\overline{\alpha}$ (A) is satisfied and it is an equality. Indeed $g(\overline{\alpha}) = 0$, i.e., $\varphi(\overline{\alpha}) = \ell(\overline{\alpha})$, that is $f(\mathbf{x}_k + \overline{\alpha}\mathbf{p}_k) = f(\mathbf{x}_k) + c_1\overline{\alpha}\nabla f(\mathbf{x}_k)^T\mathbf{p}_k$. Then

$$f(\mathbf{x}_k + \overline{\alpha}\mathbf{p}_k) - f(\mathbf{x}_k) = c_1\overline{\alpha}\nabla f(\mathbf{x}_k)^T\mathbf{p}_k. \qquad (1)$$

For the mean value theorem applied to $\varphi'(\alpha)$ in $[0, \overline{\alpha}]$, [2] it exists $\tilde{\alpha} \in (0, \overline{\alpha})$ such that

$$\varphi(\overline{\alpha}) - \varphi(0) = \overline{\alpha}\varphi'(\tilde{\alpha}),$$

[1] By assumption f is lower bounded in $\{\mathbf{x}_k + \alpha\mathbf{p}_k \mid \alpha \geq 0\}$, i.e., $\varphi(\alpha) = f(\mathbf{x}_k + \alpha\mathbf{p}_k)$ is lower bounded; then (see also Figure 3.2) it exists $\alpha > 0$, point in which $\varphi(\alpha)$ intersect the line $\ell(\alpha)$. So $g(\alpha)$ surely has a positive zero.

[2] By assumption $f \in C^1$ in $\{\mathbf{x}_k + \alpha\mathbf{p}_k \mid \alpha \geq 0\}$, i.e., $\varphi(\alpha) \in C^1([0, +\infty))$, so $\varphi'(\alpha)$ is continuous in $[0, +\infty)$, and we can apply the mean value theorem to $\varphi'(\alpha)$ in $[0, \overline{\alpha}]$.

that is

$$\overline{\alpha}\nabla f(\mathbf{x}_k+\tilde{\alpha}\mathbf{p}_k)^T\mathbf{p}_k = f(\mathbf{x}_k+\overline{\alpha}\mathbf{p}_k)-f(\mathbf{x}_k) \underbrace{=}_{(1)} c_1\underbrace{\overline{\alpha}\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}_{<0} \underbrace{>}_{0<c_1<c_2} c_2\overline{\alpha}\nabla f(\mathbf{x}_k)^T\mathbf{p}_k.$$

Deleting $\overline{\alpha}$ we obtain

$$\nabla f(\mathbf{x}_k+\tilde{\alpha}\mathbf{p}_k)^T\mathbf{p}_k > c_2\nabla f(\mathbf{x}_k)^T\mathbf{p}_k,$$

that is $\tilde{\alpha}$ satisfies (W) without equality, so it exists a neighbourhood I_W of $\tilde{\alpha}$ where (W) is satisfied. Given that $\tilde{\alpha}<\overline{\alpha}$, in $I_W\cap[0,\overline{\alpha}]\neq\varnothing$ both criterion (A) e (W) are satisfied. □

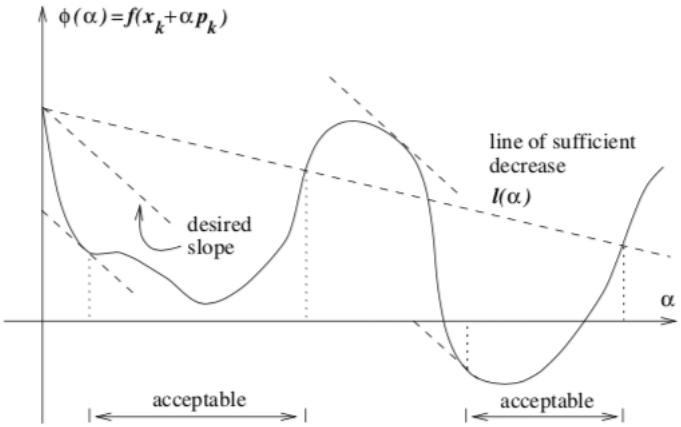

Figure 3.3: Parameters α that satisfy (A) e (W)

3.2 Convergence of line-search methods

Theorem 3.2.1. *Zoutendijk's theorem*

Let $\Omega=\{\mathbf{x}\in\mathbb{R}^n \mid f(\mathbf{x})\leq f(\mathbf{x}_0)\}$, $f\in C^1(\Omega)$ and lower bounded on Ω, $\mathbf{p}_k$ a descent direction for f, and assume that α_k satisfies (A) and (W) and that $\nabla f(\mathbf{x})$ is Lipschitz continuous in Ω.
Let ϑ_k be the angle between $-\nabla f(\mathbf{x}_k)$ and $\mathbf{p}_k$, i.e. the angle such that

$$\cos(\vartheta_k) = -\frac{\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}{\|\nabla f(\mathbf{x}_k)\|\|\mathbf{p}_k\|}.$$

The numerical series

$$\sum_{j=0}^{+\infty}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2$$

is convergent.

Proof. Adding $-\nabla f(\mathbf{x}_k)^T\mathbf{p}_k$ to both members of (W) we obtain

$$\nabla f(\mathbf{x}_k+\alpha_k\mathbf{p}_k)^T\mathbf{p}_k-\nabla f(\mathbf{x}_k)^T\mathbf{p}_k \geq c_2\nabla f(\mathbf{x}_k)^T\mathbf{p}_k-\nabla f(\mathbf{x}_k)^T\mathbf{p}_k,$$

that is

$$\begin{aligned}(c_2-1)\nabla f(\mathbf{x}_k)^T\mathbf{p}_k &\leq \big(\nabla f(\mathbf{x}_k+\alpha_k\mathbf{p}_k)-\nabla f(\mathbf{x}_k)\big)^T\mathbf{p}_k \leq \|\nabla f(\mathbf{x}_k+\alpha_k\mathbf{p}_k)-\nabla f(\mathbf{x}_k)\|\|\mathbf{p}_k\| \\ &\leq L\|(\mathbf{x}_k+\alpha_k\mathbf{p}_k)-\mathbf{x}_k\|\|\mathbf{p}_k\| = L\|\alpha_k\mathbf{p}_k\|\|\mathbf{p}_k\| = L\alpha_k\|\mathbf{p}_k\|^2,\end{aligned}$$

which gives

$$\alpha_k \geq \frac{(c_2-1)\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}{L\|\mathbf{p}_k\|^2}, \tag{3.1}$$

which is a positive amount of flow because $c_2-1<0$ and $\nabla f(\mathbf{x}_k)^T\mathbf{p}_k<0$.

Note that

$$\begin{aligned}f(\mathbf{x}_{k+1}) &\underbrace{\leq}_{\text{(A)}} f(\mathbf{x}_k)+\alpha_k\underbrace{c_1\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}_{<0} \underbrace{\leq}_{(3.1)} f(\mathbf{x}_k)+\frac{(c_2-1)c_1}{L}\frac{(\nabla f(\mathbf{x}_k)^T\mathbf{p}_k)^2}{\|\mathbf{p}_k\|^2} \\ &= f(\mathbf{x}_k)-c\frac{(\nabla f(\mathbf{x}_k)^T\mathbf{p}_k)^2}{\|\nabla f(\mathbf{x}_k)\|^2\|\mathbf{p}_k\|^2}\|\nabla f(\mathbf{x}_k)\|^2 = f(\mathbf{x}_k)-c\cos^2(\vartheta_k)\|\nabla f(\mathbf{x}_k)\|^2,\end{aligned}$$

where $c=-\dfrac{(c_2-1)c_1}{L}>0$. This holds for each α_j that satisfies the assumptions, so it holds $\forall j\leq k$:

$$f(\mathbf{x}_{j+1}) \leq f(\mathbf{x}_j)-c\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2. \tag{3.2}$$

We can then use it recursively

$$\begin{aligned}f(x_{k+1}) &\underbrace{\leq}_{\text{(3.2) with } j=k-1} f(\mathbf{x}_{k-1})-c\cos^2(\vartheta_{k-1})\|\nabla f(\mathbf{x}_{k-1})\|^2-c\cos^2(\vartheta_k)\|\nabla f(\mathbf{x}_k)\|^2\leq\dots \\ &\underbrace{\leq}_{\text{(3.2) with } j=0} f(\mathbf{x}_0)-c\sum_{j=0}^{k}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2.\end{aligned}$$

We then have

$$f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_0)-c\sum_{j=0}^{k}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2,$$

that is

$$\sum_{j=0}^{k}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2 \leq \frac{f(\mathbf{x}_0)-f(\mathbf{x}_{k+1})}{c}.$$

It holds

$$\begin{aligned}\sum_{j=0}^{+\infty}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2 &= \lim_{k\to+\infty}\sum_{j=0}^{k}\cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2 \leq \lim_{k\to+\infty}\frac{f(\mathbf{x}_0)-f(\mathbf{x}_{k+1})}{c} \\ &= \frac{f(\mathbf{x}_0)}{c}-\frac{1}{c}\lim_{k\to+\infty}f(\mathbf{x}_{k+1}) = (**).\end{aligned}$$

For the simple decrease condition (which is implied by (A)) and from the definition of Ω it holds $\mathbf{x}_k\in\Omega\ \forall k\in\mathbb{N}$. This together with the assumption that f is lower bounded in Ω, implies that $\lim_{k\to+\infty}f(\mathbf{x}_{k+1})\neq-\infty$, then

$$(**) \neq +\infty.$$

This implies that $\sum_{j=0}^{+\infty} \cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2$ is not divergent. As the series has positive terms it must converge. □

The fact that the series $\sum_{j=0}^{+\infty} \cos^2(\vartheta_j)\|\nabla f(\mathbf{x}_j)\|^2$ converges implies that

$$\lim_{k\to+\infty} \cos^2(\vartheta_k)\|\nabla f(\mathbf{x}_k)\|^2 = 0.$$

This can happen for two reasons (both or just one of them):

(i) $\lim_{k\to+\infty} \nabla f(\mathbf{x}_k) = \mathbf{0}$,

(ii) $\lim_{k\to+\infty} \cos(\vartheta_k) = 0$.

(i) In this case every accumulation point of $\{\mathbf{x}_k\}$ (if it exists) is a stationary point. Indeed, let $\widetilde{\mathbf{x}}$ be an accumulation point of $\{\mathbf{x}_k\}$, i.e., let $\widetilde{\mathbf{x}}$ be the limit point of a subsequence $\{\mathbf{x}_{k_j}\}$ of $\{\mathbf{x}_k\}$. Then

$$\nabla f(\widetilde{\mathbf{x}}) = \nabla f\left(\lim_{k_j\to+\infty} \mathbf{x}_{k_j}\right) = \lim_{k_j\to+\infty} \nabla f(\mathbf{x}_{k_j}) \underbrace{=}_{\lim_{k\to+\infty}\nabla f(\mathbf{x}_k)=\mathbf{0}} \mathbf{0}.$$

(ii) In this case, being $\cos(\vartheta_k) = -\dfrac{\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}{\|\nabla f(\mathbf{x}_k)\|\|\mathbf{p}_k\|}$, it holds

$$\lim_{k\to+\infty} \nabla f(\mathbf{x}_k)^T\mathbf{p}_k = 0,$$

that is $\nabla f(\mathbf{x}_k)$ and $\mathbf{p}_k$ tend to be orthogonal. Formally $\nabla f(\mathbf{x}_k)^T\mathbf{p}_k$ is negative, and $\mathbf{p}_k$ remains a descent direction, but actually for k large $\nabla f(\mathbf{x}_k)^T\mathbf{p}_k$ is close to 0, i.e. along the direction $\mathbf{p}_k$ the values of f are almost constant. This is a situation that can be avoided, choosing a descent direction $\mathbf{p}_k$ such that $\cos(\vartheta_k) > M$ for some $M > 0$.

With the steepest descent method, as $\mathbf{p}_k = -\nabla f(\mathbf{x}_k)$, we have

$$\cos(\vartheta_k) = -\frac{\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}{\|\nabla f(\mathbf{x}_k)\|\|\mathbf{p}_k\|} = \frac{\nabla f(\mathbf{x}_k)^T\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|\|\nabla f(\mathbf{x}_k)\|} = 1.$$

Then, under the assumptions of Zoutendijk's Theorem, it holds $\lim_{k\to+\infty} \nabla f(\mathbf{x}_k) = \mathbf{0}$, as the possibility (ii) is excluded.
With Newton's and quasi-Newton methods, because $\mathbf{p}_k = -B_k^{-1}\nabla f(\mathbf{x}_k)$ (with $B_k = H(\mathbf{x}_k)$ in Newton's method or $B_k \approx H(\mathbf{x}_k)$ for quasi-Newton method) it holds

$$\cos(\vartheta_k) = -\frac{\nabla f(\mathbf{x}_k)^T\mathbf{p}_k}{\|\nabla f(\mathbf{x}_k)\|\|\mathbf{p}_k\|} = \frac{\nabla f(\mathbf{x}_k)^T B_k^{-1}\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\| \underbrace{\|B_k^{-1}\nabla f(\mathbf{x}_k)\|}_{\le \|B_k^{-1}\|\|\nabla f(\mathbf{x}_k\|}} \ge$$

$$\ge \frac{\nabla f(\mathbf{x}_k)^T B_k^{-1}\nabla f(\mathbf{x}_k)}{\|\nabla f(\mathbf{x}_k)\|^2\|B_k^{-1}\|}.$$

To bound this we need the following definition.

Definition 3.2.1. *The Rayleigh quotient. Given $A \in \mathbb{R}^{n\times n}$ a symmetric matrix and $\mathbf{v} \in \mathbb{R}^n$ we call Rayleigh quotient associated to them the scalar*

$$r_A(\mathbf{v}) = \frac{\mathbf{v}^T A\mathbf{v}}{\|\mathbf{v}\|^2}.$$

Notably, it holds that $\forall \mathbf{v} \in \mathbb{R}^n$

$$\lambda_{\min}(A) \leq r_A(\mathbf{v}) \leq \lambda_{\max}(A).$$

We then have that

$$\begin{aligned}\cos(\vartheta_k) &= r_{B_k^{-1}}(\nabla f(\mathbf{x}_k))\frac{1}{\|B_k^{-1}\|} = r_{B_k^{-1}}(\nabla f(\mathbf{x}_k))\frac{1}{\lambda_{\max}(B_k^{-1})} = r_{B_k^{-1}}(\nabla f(\mathbf{x}_k))\lambda_{\min}(B_k)\\ &\geq \lambda_{\min}(B_k^{-1})\lambda_{\min}(B_k) = \frac{\lambda_{\min}(B_k)}{\lambda_{\max}(B_k)} = \frac{1}{k_2(B_k)}.\end{aligned}$$

Then, if it exists M such that $k_2(B_k) < M$, we have $\cos(\vartheta_k) > 1/M$ and under the assumptions of Zoutendijk's Theorem, it holds $\lim_{k\to+\infty} \nabla f(\mathbf{x}_k) = \mathbf{0}$, being again excluded the situation (ii).

3.3 Backtracking

The algorithm for a generic line-search method can be sketched as follows:

0. Given $\mathbf{x}_0, f, \text{toll}$
1. For $k = 0, 1, \ldots$
   1. Choose $\mathbf{p}_k$ descent direction (may be the steepest descent direction, Newton's direction or quasi-Newton's direction)
   2. Find α_k that satisfies (A) and (W)
   3. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$
   4. If $\|\nabla f(\mathbf{x}_{k+1})\| \leq \text{toll}$ return $\mathbf{x}_{k+1}$ (approximation of $\mathbf{x}^*$) and stop

This algorithm is well-defined because the sequence $\nabla f(\mathbf{x}_k)$ converges to 0 for Zoutendijk's Theorem and because from Wolfe's Lemma it exists α_k that satisfies (A) and (W), as required at step 2. The question is how to compute such an α_k? We can use the backtracking technique. Even if we have seen that both (A) and (W) conditions are necessary for the convergence, this technique just checks the (A) condition, we will see later why.

The *backtracking strategy* is described in the following algorithm:

0. Given $\mathbf{x}_k, \alpha_0, \mathbf{p}_k, b_{\max}, c_1 \in (0,1), \gamma \in (0,1)$
1. $\alpha_k = \alpha_0$
2. For $b = 0, 1, \ldots, b_{\max}$
   1. If $f(\mathbf{x}_k + \alpha_k \mathbf{p}_k) \leq f(\mathbf{x}_k) + \alpha_k c_1 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k$, i.e. if α_k satisfies (A), then set IND = 1 and stop,
      otherwise set $\alpha_k = \gamma \alpha_k$
3. Set IND $= -1$

This can be used at each iteration of the line-search algorithm (at step 2).

The backtracking algorithm works as follows: if $\alpha_k = \alpha_0$ does not satisfy (A), we reduce α_k by multiplication with γ and this is repeated until the new α_k satisfies (A) (the name backtracking is due to the fact that α_k if progressively reduced). In the proof of Wolfe's Lemma we have seen that it exists $\overline{\alpha} > 0$ such that each $\alpha \in [0, \overline{\alpha}]$ satisfies (A). Then we check if $\alpha_k = \alpha_0 \leq \overline{\alpha}$, if not we reduce α_k by multiplication with γ until the new $\alpha_k \leq \overline{\alpha}$. After a finite number of reductions we will obtain a α_k in $[0, \overline{\alpha}]$, so the backtracking technique never fails. However in the algorithm we decide to do at most $b_{\max}$ backtracking steps. If after $b_{\max}$ iterations $\alpha_k \leq \overline{\alpha}$ has not been found, it means that $\overline{\alpha}$ is too small, so that we will have to do really small steps in the line-search, which will lead to a really slow convergence.

It is then clear why in the algorithm we check just (A) and not (W): if $\overline{\alpha}$ is large, we get an α_k of the same order because while looking for α_k we go inside $[0, \overline{\alpha}]$ from above. If we have refused α_k because it does not satisfy (A) it means that the step is too long, than if $\gamma \alpha_k$ is accepted it cannot be too small (it is a factor γ smaller than α_k). If $\overline{\alpha}$ is small, after $b_{\max}$ backtracking steps we will not find $\alpha_k < \overline{\alpha}$. It never happens to have a too small α_k and it would therefore be redundant to check (W).
The backtracking algorithm has the flag IND in output: if IND $= 0$ it means that an α_k that satisfies both (A) and (W) has been found, otherwise if IND $= -1$ it means that after $b_{\max}$ backtracking steps α_k has not been found and so the backtracking has failed.

Each iteration of backtracking algorithm costs a function evaluation, so the algorithm costs at most $b_{\max}$ function evaluations.

In the backtracking algorithm one of the inputs is α_0, which can be chosen in various ways, depending on which directions are used in the line-search algorithm within which the backtracking is used.
If in the line-search the steepest descent direction is chosen, a popular choice is to choose α_0 by assuming that the first-order change in the function at iteration k will be the same as that obtained at the previous iteration. We then impose $\alpha_0 \nabla f(\mathbf{x}_k)^T p_k = \alpha_{k-1} \nabla f(\mathbf{x}_{k-1})^T p_{k-1}$ so that

$$\alpha_0 = \alpha_{k-1} \frac{\nabla f(\mathbf{x}_{k-1})^T p_{k-1}}{\nabla f(\mathbf{x}_k)^T p_k}$$

(α_{k-1} is the value used at the previous iteration). Another popular choice is the Barzilai-Borwein (BB) choice

$$\alpha_0 = \frac{\mathbf{s}_{k-1}^T \mathbf{s}_{k-1}}{\mathbf{s}_{k-1}^T \mathbf{y}_{k-1}}, \qquad \mathbf{s}_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}, \quad \mathbf{y}_{k-1} = \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1}),$$

that takes inspiration from the quasi-Newton methods, as we will see.
If in the line-search the Newton's or quasi-Newton's direction is chosen, we choose $\alpha_0 = 1$ (computing a Newton's step is expensive, we try to use it, if possible, to benefit from the quadratic convergence of pure Newton method).

We deduce then the algorithm for a line-search with backtracking technique:

0. Given $\mathbf{x}_0, f, k_{\max}, toll, b_{\max}, c_1 \in (0,1), \gamma \in (0,1)$
1. For $k = 0, 1, \ldots, k_{\max}$
   1. Choose $\mathbf{p}_k$ descent direction for f in $\mathbf{x}_k$
   2. Use the backtracking algorithm to find α_k
   3. If in the backtracking algorithm IND $= 0$ then set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$, otherwise stop and set IND $= -1$ (failure)
   4. If $\|\nabla f(\mathbf{x}_{k+1})\| \leq toll$ then stop, return $\mathbf{x}_{k+1}$ and IND $= 1$
2. Set IND $= -2$ (failure)

We could put the last $\mathbf{x}_k + \alpha_k \mathbf{p}_k$ and $f(\mathbf{x}_k + \alpha_k \mathbf{p}_k)$ in output of the backtracking algorithm, to save a function evaluation.
Assume that the line-search algorithm has been equipped with a proper stopping criterion (based on a tolerance toll and on a maximum number of iterations $k_{\max}$). The line-search algorithm outputs the flag IND:

- IND $= 1$ an approximation has been computed with the desired accuracy;
- IND $= -1$ the backtracking technique failed;
- IND $= -2$ the stopping criterion was not satisfied within the maximum number of iterations (failure of line-search method)

3.4 Newton's method

With the expression "Newton's method" we usually refer to the Newton's method (described in Section 2.4) plus a line-search procedure to select the step length. The resulting method is globally convergent thanks to the line-search, and in a neighbourhood of $\mathbf{x}^*$ has a quadratic rate of convergence for the following reason: thanks to the global convergence property, the sequence $\|\nabla f(\mathbf{x}_k)\|$ converges to zero. Then all the accumulation points of $\{\mathbf{x}_k\}$ are stationary points for f. If it exists an accumulation point x^* of $\{\mathbf{x}_k\}$ which is a minimizer for f, then it exists $\overline{k} > 0$ such that for each $k \geq \overline{k}$, $\mathbf{x}_k$ enters in the ball $B_\rho(\mathbf{x}^*)$, region in which we have the quadratic convergence of Newton's method. We can also prove that it exists $\tilde{k}$ such that $\alpha_k = 1$ satisfies Armijo's condition (A) for each $k \geq \tilde{k}$; i.e., starting from the iterate $\tilde{k}$, Newton's method with line-search corresponds to pure Newton's method and it then inherits its local quadratic convergence.

Chapter 4

Quasi-Newton method

Quasi-Newton methods are based on the same model of Newton's method but instead of the exact Hessian $H(\mathbf{x}_k)$ we use an SPD approximation B_k. These can be built in various ways and each choice corresponds to a different sequence of quasi-Newton directions and so to a different quasi-Newton method.
A generic quasi-Newton method is based on the following iterative scheme:

$$\left\{ \begin{array}{l} \text{Given } \mathbf{x}_0 \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k; \end{array} \right\},$$

where $\mathbf{p}_k$ is such that $B_k \mathbf{p}_k = -\nabla f(\mathbf{x}_k)$, for some approximation B_k SPD of the Hessian matrix. It is a locally convergent method that can be coupled with a line-search to become a globally convergent method and has a superlinear local rate of convergence (this is the price to be paid to avoid computation of second order derivatives).

How to compute B_k? We could approximate the Hessian of f

$$H(\mathbf{x}_k) = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1 \partial x_1}(\mathbf{x}_k) & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n}(\mathbf{x}_k) \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1}(\mathbf{x}_k) & \cdots & \frac{\partial^2 f}{\partial x_n \partial x_n}(\mathbf{x}_k) \end{pmatrix}$$

by finite differences, i.e. making this approximation

$$\frac{\partial^2 f}{\partial x_i \partial x_j}(\mathbf{x}_k) = \frac{\partial}{\partial x_j}\left(\frac{\partial f}{\partial x_i}(\mathbf{x}_k)\right) = \lim_{h \to 0} \frac{\frac{\partial f}{\partial x_i}(\mathbf{x}_k + h\mathbf{e}_j) - \frac{\partial f}{\partial x_i}(\mathbf{x}_k)}{h} \quad \approx \quad \frac{\frac{\partial f}{\partial x_i}(\mathbf{x}_k + h\mathbf{e}_j) - \frac{\partial f}{\partial x_i}(\mathbf{x}_k)}{h}$$

for each $i, j = 1, \ldots, n$. This requires n evaluations of $\nabla f(\mathbf{x})$: indeed, to build the approximations

$$\frac{\partial^2 f}{\partial x_1 \partial x_1}(\mathbf{x}_k) \approx \frac{\frac{\partial f}{\partial x_1}(\mathbf{x}_k + h\mathbf{e}_1) - \frac{\partial f}{\partial x_1}(\mathbf{x}_k)}{h}, \quad \ldots, \quad \frac{\partial^2 f}{\partial x_n \partial x_1}(\mathbf{x}_k) \approx \frac{\frac{\partial f}{\partial x_n}(\mathbf{x}_k + h\mathbf{e}_1) - \frac{\partial f}{\partial x_n}(\mathbf{x}_k)}{h}$$

it is necessary to evaluate $\nabla f(\mathbf{x})$ in $\mathbf{x}_k + h\mathbf{e}_1$, to build the approximations

$$\frac{\partial^2 f}{\partial x_1 \partial x_n}(\mathbf{x}_k) \approx \frac{\frac{\partial f}{\partial x_1}(\mathbf{x}_k + h\mathbf{e}_n) - \frac{\partial f}{\partial x_1}(\mathbf{x}_k)}{h}, \quad \ldots, \quad \frac{\partial^2 f}{\partial x_n \partial x_n}(\mathbf{x}_k) \approx \frac{\frac{\partial f}{\partial x_n}(\mathbf{x}_k + h\mathbf{e}_n) - \frac{\partial f}{\partial x_n}(\mathbf{x}_k)}{h}$$

it is necessary to evaluate $\nabla f(\mathbf{x})$ in $\mathbf{x}_k + h\mathbf{e}_n$.
Then, if we build B_k at each iteration with finite differences we need n evaluations of $\nabla f(\mathbf{x})$ for each iteration.
It is then too expensive to build a new B_k at each iteration: we need to exploit the informations obtained in the previous iterations. Assuming to have finished iteration k and to have computed B_k and $\mathbf{x}_{k+1}$, the idea of quasi-Newton methods is to avoid building an approximation B_{k+1} ex-novo and to rather obtain B_{k+1} from an update of B_k which preserves the symmetry and the positive definiteness.
The most famous quasi-Newton method is the BFGS method.

4.1 BFGS method

Let us assume to be at the end of iteration k, i.e. to have computed $\mathbf{x}_k$, B_k, α_k, $\mathbf{p}_k = -B_k^{-1}\nabla f(\mathbf{x}_k)$, $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$. We need to compute B_{k+1}. Let us built the quadratic model of f

$$m_{k+1}(\mathbf{p}) = f(\mathbf{x}_{k+1}) + \nabla f(\mathbf{x}_{k+1})^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T B_{k+1}\mathbf{p}.$$

We remark that (being $\nabla m_{k+1}(\mathbf{p}) = B_{k+1}\mathbf{p} + \nabla f(\mathbf{x}_{k+1})$) in 0 the model has the same gradient of f:

$$\nabla m_{k+1}(\mathbf{0}) = \nabla f(\mathbf{x}_{k+1}).$$

We also ask that

$$\nabla m_{k+1}(-\alpha_k \mathbf{p}_k) = \nabla f(\mathbf{x}_k)$$

i.e., that

$$\nabla m_{k+1}(\mathbf{x}_k - \mathbf{x}_{k+1}) = \nabla f(\mathbf{x}_k);$$

or

$$B_{k+1}(\mathbf{x}_k - \mathbf{x}_{k+1}) + \nabla f(\mathbf{x}_{k+1}) = \nabla f(\mathbf{x}_k).$$

If we define

$$\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k, \qquad \mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k),$$

we obtain the so-called secant equation

$$\boxed{B_{k+1}\mathbf{s}_k = \mathbf{y}_k.}$$

Anyway this equation only gives n conditions, which are not sufficient to univocally determine the n^2 coefficients of B_{k+1} (actually B_{k+1} is symmetric, so it has just $\frac{(n-1)n}{2}$ degrees of freedom, because the coefficients of the upper triangular part are equal to those of the lower triangular part). We need to impose other conditions: we ask that B_{k+1} is SPD.

Remark 1

If it exists B_{k+1} SPD that satisfies the secant equation

$$B_{k+1}\mathbf{s}_k = \mathbf{y}_k,$$

than the curvature condition is also satisfied

$$\mathbf{y}_k^T \mathbf{s}_k > 0.$$

Proof. By contradiction, let $\mathbf{y}_k^T \mathbf{s}_k \leq 0$, then

$$0 \quad \geq \quad \mathbf{y}_k^T \mathbf{s}_k \quad \underbrace{=}_{\mathbf{y}_k = B_{k+1}\mathbf{s}_k} \quad (B_{k+1}\mathbf{s}_k)^T \mathbf{s}_k \quad = \quad \mathbf{s}_k^T B_{k+1}^T \mathbf{s}_k \quad = \quad \mathbf{s}_k^T B_{k+1} \mathbf{s}_k,$$

which is impossible because B_{k+1} is positive definite by assumption. □

The curvature condition is not always satisfied for nonconvex functions, but it can be enforced imposing restrictions on the line-search, for example this is implied by the Wolfe condition:

Remark 2

If α_k satisfies Wolfe condition

$$\nabla f(\mathbf{x}_k + \alpha \mathbf{p}_k)^T \mathbf{p}_k \geq c_2 \nabla f(\mathbf{x}_k)^T \mathbf{p}_k$$

for some $c_2 \in (0,1)$, then the curvature condition $\mathbf{y}_k^T \mathbf{s}_k > 0$ is satisfied.

Proof. The proof is given in TD3. □

From Remark 1, if the curvature condition is not satisfied, then it cannot exist B_{k+1} SPD that satisfies the secant equation. In the algorithm is then advisable to verify if the curvature condition is verified. From Remark 2, we could check this condition by verifying that (W) is satisfied. However in line-search algorithm with backtrackinging technique (W) is not checked so usually we directly check the curvature condition.
These conditions are still not enough: asking that B_{k+1} is SPD imposes n additional inequalities (all principal minors must be positive); there are still some degrees of freedom left.
To determine B_{k+1} univocally, we choose B_{k+1} as the matrix, among all the symmetric matrices that satisfy the secant equation, closer (in some sense) to B_k: we ask that

$$B_{k+1} = \text{argmin}\{\|B - B_k\| \mid B = B^T, B\mathbf{s}_k = \mathbf{y}_k\},$$

where we know that B_k is SPD and that the curvature condition $\mathbf{y}_k^T \mathbf{s}_k > 0$ is satisfied.
To solve this problem we can choose various matrix norms, and each of them leads to a different quasi-Newton method. The norm that makes the solution easier is the weighted Frobenius norm[1] with weight W SPD such that $W\mathbf{y}_k = \mathbf{s}_k$. For example we can choose, assuming $H(x)$

[1] Let $A \in \mathbb{R}^{n\times n}$. The Frobenius norm of A is defined as

$$\|A\|_F = \sqrt{\sum_{i,j=1}^{n} a_{ij}^2}.$$

Given $W \in \mathbb{R}^{n\times n}$ SPD, we know that

$$W = O^T \Lambda O \qquad \text{for some } O \in \mathbb{R}^{n\times n} \text{ orthogonal, } \Lambda = \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{pmatrix}, \ \lambda_1, \dots, \lambda_n > 0,$$

and so we can define

$$\sqrt{W} = O^T \sqrt{\Lambda} O, \qquad \sqrt{\Lambda} = \begin{pmatrix} \sqrt{\lambda_1} & & \\ & \ddots & \\ & & \sqrt{\lambda_n} \end{pmatrix},$$

to be positive definite in $[x_k, x_k + p_k]$,

$$W = \bar{G}_k^{-1},$$

where $\bar{G}_k$ is the average Hessian matrix

$$\bar{G}_k = \int_0^1 H(\mathbf{x}_k + t\alpha_k \mathbf{p}_k)\, \mathrm{d}t. \tag{4.1}$$

W is such that $W\mathbf{y}_k = \mathbf{s}_k$ because $\bar{G}_k^{-1}\mathbf{y}_k = \mathbf{s}_k$ as

$$\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) = \Big[\nabla f(\mathbf{x}_k + t(\mathbf{x}_{k+1} - \mathbf{x}_k))\Big]_{t=0}^{t=1} =$$

$$= \int_0^1 H(\mathbf{x}_k + t(\mathbf{x}_{k+1} - \mathbf{x}_k))(\mathbf{x}_{k+1} - \mathbf{x}_k)\, \mathrm{d}t = \int_0^1 H(\mathbf{x}_k + t\alpha_k \mathbf{p}_k)\, \mathrm{d}t\, \mathbf{s}_k = \bar{G}_k \mathbf{s}_k.$$

With this choice, the unique solution of the problem is

$$B_{k+1} = (I - \rho_k \mathbf{y}_k \mathbf{s}_k^T) B_k (I - \rho_k \mathbf{s}_k \mathbf{y}_k^T) + \rho_k \mathbf{y}_k \mathbf{y}_k^T, \tag{DFP}$$

where

$$\rho_k = \frac{1}{\mathbf{y}_k^T \mathbf{s}_k} > 0.$$

We can prove that if B_k is SPD then also B_{k+1} is SPD.
This formula is called DFP because it was discovered empirically by physician Davidon in 1959, and in 1963 the mathematicians Fletcher and Powell explained rigorously why this update technique works: they understood that B_{k+1} was the solution of this minimum problem.
DFP is an update formula, because it allows to build B_{k+1} from B_k, $\nabla f(\mathbf{x}_{k+1})$ and $\nabla f(\mathbf{x}_k)$: the basic idea of quasi-Newton methods is indeed that of avoiding building ex-novo a matrix B_{k+1} to approximate $H(\mathbf{x}_{k+1})$, and rather to obtain B_{k+1} updating B_k (preserving the symmetry and the positive definiteness) by using the informations $\nabla f(\mathbf{x}_{k+1})$ and $\nabla f(\mathbf{x}_k)$.
Because

$$\begin{aligned} B_{k+1} &= (I - \rho_k \mathbf{y}_k \mathbf{s}_k^T) B_k (I - \rho_k \mathbf{s}_k \mathbf{y}_k^T) + \rho_k \mathbf{y}_k \mathbf{y}_k^T \\ &= B_k - \rho_k \mathbf{y}_k \mathbf{s}_k^T B_k - \rho_k B_k \mathbf{s}_k \mathbf{y}_k^T + \rho_k^2 \mathbf{y}_k \mathbf{s}_k^T B_k \mathbf{s}_k \mathbf{y}_k^T + \rho_k \mathbf{y}_k \mathbf{y}_k^T \end{aligned}$$

is obtained from B_k by adding rank 1 matrices (globally is we obtain B_{k+1} with a rank 2 modification of B_k), we can use the Sherman-Morrison-Woodbury formula [2] to compute B_{k+1}^{-1}

---

and it holds

$$\sqrt{W}\sqrt{W} = O^T\sqrt{\Lambda}OO^T\sqrt{\Lambda}O = O^T\sqrt{\Lambda}\sqrt{\Lambda}O = O^T\Lambda O = W.$$

Given $W \in \mathbb{R}^{n\times n}$ SPD, the weighted Frobenius of A with weight W is defined as

$$\|A\|_W = \|\sqrt{W}A\sqrt{W}\|_F.$$

[2] Let $A \in \mathbb{R}^{n\times n}$ be invertible. Let $\overline{A}$ a matrix obtained from A by addition of a rank 1 matrix

$$\overline{A} = A + \mathbf{a}\mathbf{b}^T$$

with $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$. If $\overline{A}$ is invertible, then we can compute $\overline{A}^{-1}$ from A^{-1} and by doing just some matrix-vector products:

$$\overline{A}^{-1} = A^{-1} - \frac{A^{-1}\mathbf{a}\mathbf{b}^T A^{-1}}{1 + \mathbf{b}^T A^{-1}\mathbf{a}}.$$

from B_k^{-1}:

$$B_{k+1}^{-1} = B_k^{-1} - \frac{B_k^{-1}\mathbf{y}_k\mathbf{y}_k^T B_k^{-1}}{\mathbf{y}_k^T B_k^{-1}\mathbf{y}_k} + \frac{\mathbf{s}_k\mathbf{s}_k^T}{\mathbf{y}_k^T\mathbf{s}_k}.$$

If we build the sequence B_k starting from a matrix B_0 of which the inverse B_0^{-1} is known, then using the prevoius formula, we can compute all the B_k^{-1} by performing just matrix-vector products, and we can also compute $\mathbf{p}_k$ by a matrix-vector product $\mathbf{p}_k = -B_k^{-1}\nabla f(\mathbf{x}_k)$ and we do not need to solve the linear system $B_k\mathbf{p}_k = -\nabla f(\mathbf{x}_k)$.

The DFP updating formula is quite effective, but it was soon superseded by the BFGS formula, which is presently considered to be the most effective of all quasi-Newton updating formulae. It was proposed by Broyden, Fletcher, Goldfarb, Shanno. The BFGS update formula instead of approximating the Hessian matrix, imposes analogous conditions on approximations $\tilde{B}_k$ of the inverse of the Hessian matrix.

Let us assume to be at the and of iteration k, we have computed $\mathbf{x}_k$, $\tilde{B}_k$, α_k, $\mathbf{p}_k = -\tilde{B}_k\nabla f(\mathbf{x}_k)$, $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k\mathbf{p}_k$. We need to compute $\tilde{B}_{k+1}$. To determine $\tilde{B}_{k+1}$ univocally, we ask $\tilde{B}_{k+1}$ to be, among all the symmetric matrices that satisfy the equation $\tilde{B}\mathbf{y}_k = \mathbf{s}_k$, the closest matrix to $\tilde{B}_k$:

$$\tilde{B}_{k+1} = \text{argmin}\{\|\tilde{B} - \tilde{B}_k\| \mid \tilde{B} = \tilde{B}^T, H\mathbf{y}_k = \mathbf{s}_k\},$$

where we know that $\tilde{B}_k$ is SPD and the curvature condition $\mathbf{y}_k^T\mathbf{s}_k > 0$ holds.
We choose again the Frobenius norm with weight W such that $W\mathbf{s}_k = \mathbf{y}_k$. Also in this case we can choose $W = \bar{G}_k$ (it holds $W\mathbf{s}_k = \mathbf{y}_k$). With this choice, the unique solution of the minimization problem is

$$\tilde{B}_{k+1} = (I - \rho_k\mathbf{s}_k\mathbf{y}_k^T)\tilde{B}_k(I - \rho_k\mathbf{y}_k\mathbf{s}_k^T) + \rho_k\mathbf{s}_k\mathbf{s}_k^T, \qquad \text{(BFGS)}$$

where

$$\rho_k = \frac{1}{\mathbf{y}_k^T\mathbf{s}_k} > 0.$$

We can prove that if $\tilde{B}_k$ is SPD then also $\tilde{B}_{k+1}$ is SPD.

We can derive a version of the BFGS algorithm that works with the Hessian approximation B_k rather than its inverse $\tilde{B}_k$. The update formula for B_k is obtained by simply applying the Sherman-Morrison-Woodbury formula to obtain (BFGS)

$$B_{k+1} = B_k - \frac{B_k\mathbf{s}_k\mathbf{s}_k^T B_k}{\mathbf{s}_k^T B_k\mathbf{s}_k} + \frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\mathbf{s}_k}. \tag{4.2}$$

The last question is how to chose $\tilde{B}_0$? Unlucky there is not a general formula that works well in all cases. Usually we choose $\tilde{B}_0 = I_n$, or $\tilde{B}_0 = \gamma I_n$ with $\gamma = \frac{\mathbf{y}_0^T\mathbf{s}_0}{\mathbf{y}_0^T\mathbf{y}_0}$, or $\tilde{B}_0 = B_0^{-1}$ after having computed an approximation B_0 of $H(\mathbf{x}_0)$ by finite differences.

Each iteration of quasi-Newton methods can be performed at a cost of $O(n^2)$ arithmetic operations (plus the cost of function and gradient evaluations); there are no $O(n^3)$ operations such as linear system solves or matrix-matrix operations. The algorithm is robust, and its rate of convergence is superlinear, which is fast enough for most practical purposes. Even though Newton's method converges more rapidly (that is, quadratically), its cost per iteration is higher

because it requires the solution of a linear system. A more important advantage for BFGS is, of course, that it does not require calculation of second derivatives.

We describe the algorithm of the BFGS method.

Given $\mathbf{x}_0$, $\epsilon > 0$, inverse Hessian approximation $\tilde{B}_0$, set $k = 0$.

While $\|\nabla f(\mathbf{x}_k)\| > \epsilon$

1. Compute the search direction $\mathbf{p}_k = -\tilde{B}_k \nabla f(\mathbf{x}_k)$.
2. Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k$ where α_k is computed from a line search procedure to satisfy (A)+(W)
3. Define $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$ and $\mathbf{y}_k = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$
4. Compute $\tilde{B}_{k+1}$ by means of ??
5. Set $k = k + 1$

4.2 Global convergence of the BFGS method

We study the global convergence of BFGS, with a practical line search, when applied to a smooth convex function from an arbitrary starting point x_0 and from any initial Hessian approximation B_0 that is symmetric and positive definite.

We assume the following assumption

Assumption 4.2.1. *We assume that*

1. *The objective function f is twice continuously differentiable.*
2. *The level set $\mathcal{L} = \{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ is convex, and there exist positive constants m and M such that*
$$m\|z\|^2 \leq z^T H(\mathbf{x}) z \leq M\|z\|^2 \tag{4.3}$$
for all $\mathbf{z} \in \mathbb{R}^n$ and $\mathbf{x} \in \mathcal{L}$.

The second part of this assumption implies that $H(x)$ is positive definite on $\mathcal{L}$ and that f has a unique minimizer x^*. We have seen that $\mathbf{y}_k = \bar{G}_k s_k$, where $\bar{G}_k$ is the average Hessian defined in (4.1). From this and (4.3) we obtain

$$\frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{s}_k} = \frac{\mathbf{s}_k^T \bar{G}_k \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{s}_k} \geq m. \tag{4.4}$$

From the assumption $\bar{G}_k$ is positive definite, so its square root is well-defined. Therefore, we have by defining $\mathbf{z}_k = \bar{G}_k^{1/2}$ that

$$\frac{\mathbf{y}_k^T \mathbf{y}_k}{\mathbf{y}_k^T \mathbf{s}_k} = \frac{\mathbf{z}_k^T \bar{G}_k \mathbf{z}_k}{\mathbf{z}_k^T \mathbf{z}_k} \leq M. \tag{4.5}$$

We are now ready to present the global convergence result for the BFGS method.

Theorem 4.2.1. *Let B_0 be any symmetric positive definite initial matrix, and let x_0 be a starting point for which Assumption 4.2.1 is satisfied. Then the sequence $\{xk\}$ generated by BFGS Algorithm converges to the minimizer x^* of f .*

Proof. Some points of the proof (marked by "see TD") are treated in Exercise 3 of TD 4.

Let us define

$$m_k = \frac{\mathbf{y}_k^T \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{s}_k}, \qquad M_k = \frac{\mathbf{y}_k^T \mathbf{y}_k}{\mathbf{y}_k^T \mathbf{s}_k} \tag{4.6}$$

and note from (4.4) and (4.5) that

$$m_k \geq m, M_k \leq M. \tag{4.7}$$

By computing the trace of the BFGS approximation (4.2), we obtain that (see TD)

$$\mathrm{trace}(\mathrm{B_{k+1}}) = \mathrm{trace}(\mathrm{B_k}) - \frac{\|\mathrm{B_k s_k}\|^2}{\mathbf{s}_\mathrm{k}^\mathrm{T}\mathrm{B_k s_k}} + \frac{\|\mathbf{y_k}\|^2}{\mathbf{y}_\mathrm{k}^\mathrm{T}\mathbf{s}_\mathrm{k}}. \tag{4.8}$$

We can also show (see TD) that

$$\det(\mathrm{B_{k+1}}) = \det(\mathrm{B_k})\frac{\mathbf{y}_\mathrm{k}^\mathrm{T}\mathbf{s}_\mathrm{k}}{\mathbf{s}_\mathrm{k}^\mathrm{T}\mathrm{B_k s_k}}. \tag{4.9}$$

Let us also define

$$\cos(\theta_k) = \frac{\mathbf{s}_k^T B_k \mathbf{s}_k}{\|\mathbf{s}_k\|\|B_k \mathbf{s}_k\|}, \quad q_k = \frac{\mathbf{s}_k^T B_k \mathbf{s}_k}{\mathbf{s}_k^T \mathbf{s}_k}, \tag{4.10}$$

so that θ_k is the angle between $\mathbf{s}_k$ and $B_k\mathbf{s}_k$. We then obtain that

$$\frac{\|B_k \mathbf{s}_k\|^2}{\mathbf{s}_k^T B_k \mathbf{s}_k} = \frac{\|B_k \mathbf{s}_k\|^2 \|\mathbf{s}_k\|^2}{(\mathbf{s}_k^T B_k \mathbf{s}_k)^2} \frac{\mathbf{s}_k^T B_k \mathbf{s}_k}{\|\mathbf{s}_k\|^2} = \frac{q_k}{\cos^2(\theta_k)}. \tag{4.11}$$

In addition, we have from (4.6) that

$$\det(\mathrm{B_{k+1}}) = \det(\mathrm{B_{k+1}})\frac{\mathbf{y}_\mathrm{k}^\mathrm{T}\mathbf{s}_\mathrm{k}}{\mathbf{s}_\mathrm{k}^\mathrm{T}\mathbf{s}_\mathrm{k}} \frac{\mathbf{s}_\mathrm{k}^\mathrm{T}\mathbf{s}_\mathrm{k}}{\mathbf{s}_\mathrm{k}^\mathrm{T}\mathrm{B_k s_k}} = \det(\mathrm{B_{k+1}})\frac{\mathrm{m_k}}{\mathrm{q_k}}. \tag{4.12}$$

We now combine the trace and determinant by introducing the following function of a positive definite matrix B:

$$\psi(B) = \mathrm{trace(B)} - \ln(\det(\mathrm{B})) \tag{4.13}$$

where $\ln(\cdot)$ denotes the natural logarithm. It is not difficult to show that $\psi(B) > 0$ (see TD). By using (4.6) and (4.8)-(4.13), we have that

$$\begin{aligned}
\psi(B_{k+1}) =& \psi(B_k) + M_k - \frac{q_k}{\cos^2(\theta_k)} - \ln(\det(\mathrm{B_k})) - \ln \mathrm{m_k} + \ln \mathrm{q_k} \\
=& \psi(B_k) + (M_k - \ln(m_k) - 1) \\
&+ \left[1 - \frac{q_k}{\cos^2(\theta_k)} + \ln\left(\frac{q_k}{\cos^2(\theta_k)}\right)\right] + \ln(\cos^2(\theta_k))
\end{aligned} \tag{4.14}$$

Now, since the function $h(t) = 1 - t + \ln(t) \leq 0$ is nonpositive for all $t > 0$ (see TD), the term inside the square brackets is nonpositive, and thus from (4.7) and (4.14) we have

$$0 < \psi(B_{k+1}) - \psi(B_1) + ck + \sum_{j=1}^{k} \ln(\cos^2(\theta_j)) \tag{4.15}$$

where we can assume the constant $c = M - \ln(m) - 1$ to be positive, without loss of generality. We now relate these expressions to the results given in Chapter 3. Note from the form $s_k = -\alpha_k B_k^{-1} \nabla f_k$ the quasi-Newton iteration that $\cos(\theta_k)$ defined by (4.10) is the angle between the steepest descent direction and the search direction, which plays a crucial role in the global convergence theory of Chapter 3. From the result of Zoutendijk's theorem we know that the sequence $\|\nabla f(\mathbf{x}_k)\|$ generated by the line search algorithm is bounded away from zero only if $\cos\theta_j \to 0$. Let us then proceed by contradiction and assume that $\cos\theta_j \to 0$. Then there exists $k_1 > 0$ such that for all $j > k_1$, we have

$$\ln(\cos^2(\theta_j)) < -2c,$$

where c is the constant defined above. Using this inequality in (4.15) we find the following relations to be true for all $k > k_1$:

$$\begin{aligned} 0 &< \psi(B_1) + ck + \sum_{j=1}^{k_1} \ln(\cos^2(\theta_j)) + \sum_{j=k_1+1}^{k} (-2c) \\ &= \psi(B_1) + \sum_{j=1}^{k_1} \ln(\cos^2(\theta_j)) + 2ck_1 - ck. \end{aligned}$$

However, the right-hand-side is negative for large k, giving a contradiction. Therefore, there exists a subsequence of indices $\{j_k\}$ such that $\{\cos(\theta_{j_k})\} \geq \delta > 0$. By Zoutendijk's theorem this limit implies that $\liminf \|\nabla f(\mathbf{x}_k)\| \to 0$. Since the problem is strongly convex, the latter limit is enough to prove that $x_k \to x^*$. □

Chapter 5

Nonlinear least-squares problems

5.1 Background: modelling, regression

Nonlinear least-squares problems often arise when we want to fit a model to some data, i.e., when we solve a regression problem.

Assume to have some experimental measurements

$$(t_i, y_i), \qquad i = 1, ..., m.$$

The measurements are some (usually noisy) realisations of a function $y : \mathbb{R} \longrightarrow \mathbb{R}$ of the variable t that describes the observed phenomenon. We want to approximate this function by a model $m(\mathbf{x}; t)$, where $\mathbf{x} = (x_1, \ldots, x_n)^T$ are some parameters to be determined. We can choose the value of these parameters to adapt the model to the data at best. [1]
The true variables of the model are then the parameters $\mathbf{x}$, the variable t of the original function will take the measured values.

To find the best parameters we try to minimize the distance of the model from the measurements, i.e., we look for $\mathbf{x} \in \mathbb{R}^n$ that minimizes the amount of flow

$$\frac{1}{2} \left\| \begin{pmatrix} m(\mathbf{x}; t_1) - y_1 \\ \vdots \\ m(t_m, \mathbf{x}) - y_m \end{pmatrix} \right\|^2 = \frac{1}{2} \sum_{i=1}^{m} \big(m(\mathbf{x}; t_i) - y_i\big)^2.$$

This is a nonlinear (if m is nonlinear with respect to $\mathbf{x}$) least-squares problem.

5.2 General concepts

Let

$$\begin{array}{rccc} F: & \mathbb{R}^n & \longrightarrow & \mathbb{R}^m \\ & \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} & \longmapsto & \mathbf{F}(\mathbf{x}) = \begin{pmatrix} F_1(\mathbf{x}) \\ \vdots \\ F_m(\mathbf{x}) \end{pmatrix} \end{array}$$

[1] Usually we choose a family of functions to which the model belongs, which is parametrized by some parameters. For example we can have an exponential model $m(\mathbf{x}; t) = x_1 e^{x_2 t}$, with $\mathbf{x} = (x_1, x_2)^T$ or a linear model $m(\mathbf{x}; t) = x_1 + x_2 t$.

and

$$\begin{array}{rccl} f: & \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & \mathbf{x} & \longmapsto & f(\mathbf{x}) = \dfrac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2 = \dfrac{1}{2}\displaystyle\sum_{i=1}^{m} F_i(\mathbf{x})^2. \end{array}$$

The general form of a nonlinear least-squares problem is

$$\min_{\mathbf{x}} f(\mathbf{x}) = \frac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2.$$

In the following we will assume, as it is common in applications, that $m \geq n$.

Let $\mathbf{x} \in \mathbb{R}^n$. If F is differentiable in $\mathbf{x}$ (i.e., if $F_1, \ldots, F_m$ are differentiable in $\mathbf{x}$), the Jacobain matrix of F in $\mathbf{x}$ is $J(\mathbf{x}) \in \mathbb{R}^{m \times n}$

$$J(\mathbf{x}) = \begin{pmatrix} \nabla F_1(\mathbf{x})^T \\ \vdots \\ \nabla F_m(\mathbf{x})^T \end{pmatrix} = \begin{pmatrix} \dfrac{\partial F_1}{\partial x_1}(\mathbf{x}) & \cdots & \dfrac{\partial F_1}{\partial x_n}(\mathbf{x}) \\ \vdots & & \vdots \\ \dfrac{\partial F_m}{\partial x_1}(\mathbf{x}) & \cdots & \dfrac{\partial F_m}{\partial x_n}(\mathbf{x}) \end{pmatrix}.$$

Let $\mathbf{x}^*$ be a solution of the least-squares problem. The residual is the following amount of flow:

$$r = f(\mathbf{x}^*) = \frac{1}{2}\|\mathbf{F}(\mathbf{x}^*)\|^2.$$

If it exists a solution for which $r = 0$ we say that the problem is a zero residual problem.

Remark 5.2.1. *It holds*

(a) $\nabla f(\mathbf{x}) = J(\mathbf{x})^T \mathbf{F}(\mathbf{x})$,

(b) $H(\mathbf{x}) = J(\mathbf{x})^T J(\mathbf{x}) + \displaystyle\sum_{i=1}^{m} F_i(\mathbf{x}) H_{F_i}(\mathbf{x})$.

Proof. (a) $\forall j = 1, \ldots, n$

$$\begin{aligned} \Big(\nabla f(\mathbf{x})\Big)_j &= \frac{\partial}{\partial x_j} f(\mathbf{x}) = \frac{\partial}{\partial x_j} \frac{1}{2} \sum_{i=1}^{m} F_i(\mathbf{x})^2 = \frac{1}{2} \sum_{i=1}^{m} \frac{\partial F_i(\mathbf{x})^2}{\partial x_j} = \frac{1}{2} \sum_{i=1}^{m} 2F_i(\mathbf{x}) \frac{\partial F_i(\mathbf{x})}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial F_i(\mathbf{x})}{\partial x_j} F_i(\mathbf{x}) \\ &= \sum_{i=1}^{m} \Big(J(\mathbf{x})\Big)_{ij} \Big(\mathbf{F}(\mathbf{x})\Big)_i = \sum_{i=1}^{m} \Big(J(\mathbf{x})^T\Big)_{ji} \Big(\mathbf{F}(\mathbf{x})\Big)_i = \Big(J(\mathbf{x})^T \mathbf{F}(\mathbf{x})\Big)_j. \end{aligned}$$

(b) $\forall j, k = 1, \ldots, n$

$$\begin{aligned} \Big(H(\mathbf{x})\Big)_{jk} &= \frac{\partial^2 f(\mathbf{x})}{\partial x_j \partial x_k} = \frac{\partial}{\partial x_k} \frac{\partial f(\mathbf{x})}{\partial x_j} = \frac{\partial}{\partial x_k} \Big(\nabla f(\mathbf{x})\Big)_j \underbrace{=}_{\text{(a)}} \\ &= \frac{\partial}{\partial x_k} \sum_{i=1}^{m} F_i(\mathbf{x}) \frac{\partial F_i(\mathbf{x})}{\partial x_j} = \sum_{i=1}^{m} \frac{\partial F_i(\mathbf{x})}{\partial x_k} \frac{\partial F_i(\mathbf{x})}{\partial x_j} + \sum_{i=1}^{m} F_i(\mathbf{x}) \frac{\partial^2 F_i(\mathbf{x})}{\partial x_j \partial x_k} \\ &= \sum_{i=1}^{m} \Big(J(\mathbf{x})\Big)_{ik} \Big(J(\mathbf{x})\Big)_{ij} + \sum_{i=1}^{m} F_i(\mathbf{x}) \Big(H_{F_i}(\mathbf{x})\Big)_{jk}. \end{aligned}$$

Because

$$\sum_{i=1}^{m} \Big(J(\mathbf{x})\Big)_{ik}\Big(J(\mathbf{x})\Big)_{ij} = \sum_{i=1}^{m} \Big(J(\mathbf{x})^T\Big)_{ki}\Big(J(\mathbf{x})\Big)_{ij} = \Big(\underbrace{J(\mathbf{x})^T J(\mathbf{x})}_{\text{is symmetric}}\Big)_{kj} = \Big(J(\mathbf{x})^T J(\mathbf{x})\Big)_{jk}$$

it holds

$$\Big(H(\mathbf{x})\Big)_{jk} = \Big(J(\mathbf{x})^T J(\mathbf{x}\Big)_{jk} + \sum_{i=1}^{m} F_i(\mathbf{x})\Big(H_{F_i}(\mathbf{x})\Big)_{jk}.$$

□

Remark that $r = 0$ means $\mathbf{F}(\mathbf{x}^*) = \mathbf{0}$, i.e., $\mathbf{x}^*$ is the solution of the nonlinear system

$$\mathbf{F}(\mathbf{x}) = \mathbf{0}.$$

If $r = 0$, then

$$H(\mathbf{x}^*) \quad = \quad J(\mathbf{x}^*)^T J(\mathbf{x}^*) + \sum_{i=1}^{m} \underbrace{F_i(\mathbf{x}^*)}_{=0} H_{F_i}(\mathbf{x}^*) \quad = \quad J(\mathbf{x}^*)^T J(\mathbf{x}^*).$$

In many situations the term $J(\mathbf{x})^T J(\mathbf{x})$ is a good approximation of the Hessian matrix $H(\mathbf{x})$, for example when the residuals are small ($F_i(\mathbf{x}) \approx 0$) or when the model is almost linear ($H_{F_i}(\mathbf{x}) \approx 0$). In such cases, for $\mathbf{x}$ close to $\mathbf{x}^*$, we can use the following approximation:

$$H(\mathbf{x}) = J(\mathbf{x})^T J(\mathbf{x}) + \sum_{i=1}^{m} F_i(\mathbf{x}) H_{F_i}(\mathbf{x}) \approx J(\mathbf{x})^T J(\mathbf{x}).$$

This is convenient because it allows us to get a reliable approximation to the second order derivatives by employing just first order derivatives, that can also be used to compute $\nabla f(x)$.

5.3 Linear least-squares problems

In the special case in which each function F_i is linear, we have $\mathbf{F}(\mathbf{x}) = A\mathbf{x} + b$, the Jacobian A is constant, and we can write

$$f(x) = \frac{1}{2}\|A\mathbf{x} + \mathbf{b}\|^2.$$

We also have

$$\nabla f(\mathbf{x}) = A^T(A\mathbf{x} + \mathbf{b}) \qquad\qquad H(\mathbf{x}) = A^T A.$$

The second term in (b) disappears as the function is linear so $H_{F_i} = 0$ for all i. Function f is always convex and any solution must satisfy $\nabla f(x) = A^T(A\mathbf{x} + \mathbf{b}) = 0$, which leads to the *normal equations*:

$$A^T A\mathbf{x} = -A^T\mathbf{b}.$$

5.4 Algorithms for nonlinear least-squares problems

5.4.1 Gauss-Newton method

Let us assume that $r \approx 0$, that $J(\mathbf{x}_k) \in \mathbb{R}^{m\times n}$ with $m \geq n$, and that $J(\mathbf{x}_k)$ is a full rank matrix, i.e., $\text{rk}(J(\mathbf{x}_k)) = n$. The Gauss-Newton direction $\mathbf{p}_k^{GN}$ is the quasi-Newton direction obtained choosing

$$B_k = J(\mathbf{x}_k)^T J(\mathbf{x}_k),$$

i.e., as $\nabla f(\mathbf{x}_k) = J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k)$, $\mathbf{p}_k^{GN}$ is the solution of the quasi-Newton system

$$J(\mathbf{x}_k)^T J(\mathbf{x}_k)\mathbf{p} = -J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k). \tag{5.1}$$

The matrix $J(\mathbf{x}_k)^T J(\mathbf{x}_k)$ is SPD when $J(\mathbf{x}_k)$ has full rank (all the eigenvalues are nonzero). Indeed $\forall \mathbf{v} \in \mathbb{R}^n$, $v \neq 0$

$$\mathbf{v}^T J(\mathbf{x}_k)^T J(\mathbf{x}_k)\mathbf{v} \quad = \quad \big(J(\mathbf{x}_k)\mathbf{v}\big)^T J(\mathbf{x}_k)\mathbf{v} \quad = \quad \|J(\mathbf{x}_k)\mathbf{v}\|^2 \quad > \quad 0.$$

$\mathbf{p}_k^{GN}$ is a descent direction for f in $\mathbf{x}_k$ because

$$\begin{aligned}\nabla f(\mathbf{x}_k)^T \mathbf{p}_k^{GN} &= \big(J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k)\big)^T \mathbf{p}_k^{GN} = \Big(- J(\mathbf{x}_k)^T J(\mathbf{x}_k)\mathbf{p}_k^{GN}\Big)^T \mathbf{p}_k^{GN} \\ &= -(\mathbf{p}_k^{GN})^T J(\mathbf{x}_k)^T J(\mathbf{x}_k)\mathbf{p}_k^{GN} = -\|J(\mathbf{x}_k)\mathbf{p}_k^{GN}\|^2 < 0,\end{aligned}$$

where this last inequality follows from the fact that $J(\mathbf{x}_k)^T J(\mathbf{x}_k)$ is positive definite.
If $J(\mathbf{x}_k)$ is not full rank, then $J(\mathbf{x}_k)^T J(\mathbf{x}_k)$ is still symmetric but it is just positive semidefinite. In this case a possibility is to choose $B_k = J(\mathbf{x}_k)^T J(\mathbf{x}_k) + \varepsilon I_n$ with $\varepsilon > 0$ small, so that B_k is SPD[2] and it approximates well $J(\mathbf{x}_k)^T J(\mathbf{x}_k)$ (because ε is small). We will discuss this in Section **??**.

The Gauss-Newton method can be also derived by approximating the objective function F by a linear model at each iteration: $\mathbf{F}(\mathbf{x}_k + \mathbf{p}) \sim \mathbf{F}(\mathbf{x}_k) + J(\mathbf{x}_k)\mathbf{p}$. Using as a model for f the squared norm of the linear model of F, we obtain a quadratic approximation to f with approximated Hessian:

$$\begin{aligned}m_k^{GN}(\mathbf{p}) = \frac{1}{2}\|\mathbf{F}(\mathbf{x}_k) + J(\mathbf{x}_k)p\|^2 &= \frac{1}{2}\|\mathbf{F}(\mathbf{x}_k)\|^2 + J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k)\mathbf{p} + \frac{1}{2}\mathbf{p}^T J(\mathbf{x}_k)^T J(\mathbf{x}_k)^T \mathbf{p} \\ &= f(\mathbf{x}_k) + \nabla f(\mathbf{x}_k)^T \mathbf{p} + \frac{1}{2}\mathbf{p}^T J(\mathbf{x}_k)^T J(\mathbf{x}_k)^T \mathbf{p}.\end{aligned}$$

To get the step at each iteration we minimize the model, which amounts to solve a linear least-squares problem, whose normal equations are exactly (5.1). By using the SVD decomposition of $J(\mathbf{x}_k) = USV^T$, we can write (see TD 5) the solution of this problem as

$$\mathbf{x}^* = \sum_{i=1}^{n} \frac{\mathbf{u}_i^T \mathbf{F}}{\sigma_i}\mathbf{v}_i.$$

[2] If $A \in \mathbb{R}^{n\times n}$ is positive semidefinite, then $\forall \varepsilon > 0$ $A + \varepsilon I_n$ is positive definite.
Indeed, let $\lambda_1, \ldots, \lambda_n$ be the eigenvalues of A and $\mathbf{v}_1, \ldots, \mathbf{v}_n$ the relative eigenvectors. $\forall i = 1, \ldots, n$

$$(A + \varepsilon I_n)\mathbf{v}_i = A\mathbf{v}_i + \varepsilon \mathbf{v}_i = \lambda_i \mathbf{v}_i + \varepsilon \mathbf{v}_i = (\lambda_i + \varepsilon)\mathbf{v}_i,$$

i.e., $\mathbf{v}_i$ is an eigenvector of $A + \varepsilon I_n$ with eigenvalue $\lambda_i + \varepsilon$. Then the eigenvalues of $A + \varepsilon I_n$ are $\lambda_1 + \varepsilon, \ldots, \lambda_n + \varepsilon$, which are all positive because $\lambda_1, \ldots, \lambda_n \geq 0$ as A is positive semidefinite.

Gauss-Newton method is then based on the following iterative scheme:

$$\left\{ \begin{array}{l} \text{Given } \mathbf{x}_0 \\ \mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{p}_k^{GN} \end{array} \right\}.$$

The convergence of the method clearly depends on how important is the term we have discarded in the Hessian approximation. As shown in the following theorem, the method is locally convergent with quadratic local convergence in case of zero residual. If the residual is nonzero, if $\left\| \sum_{i=1}^{m} F_i(\mathbf{x}^*) H_{F_i}(\mathbf{x}^*) \right\|$ is small with respect to the smallest eigenvalue of $J(x^*)^T J(x^*)$, then the convergence is linear. Otherwise, there is no guarantee of convergence for the method.

Theorem 5.4.1. *Let $F : \mathbb{R}^n \to \mathbb{R}^m$ and let $f(x) = \frac{1}{2}\|\mathbf{F}(\mathbf{x})\|^2$ be twice continuously differentiable in an open convex set $\mathcal{D} \subset \mathbb{R}^n$. Assume that $J(\mathbf{x})$ is Lipschitz continuous in $\mathcal{D}$ with Lipschitz constant γ and that $\|J(\mathbf{x})\| \leq \alpha$ for all $\mathbf{x} \in \mathcal{D}$. Assume that there exists $\mathbf{x}^* \in \mathcal{D}$ such that $J(\mathbf{x}^*)^T \mathbf{F}(\mathbf{x}^*) = 0$. Let λ be the smallest eigenvalue of $J(\mathbf{x}^*)^T J(\mathbf{x}^*)$ and assume that*

$$\|(J(\mathbf{x}) - J(\mathbf{x}^*))^T \mathbf{F}(\mathbf{x}^*)\| \leq \sigma \|\mathbf{x} - \mathbf{x}_*\|$$

for some constant $\sigma \geq 0$ and for all $\mathbf{x} \in \mathcal{D}$. If $\sigma < \lambda$ then for any $c \in (1, \lambda/\sigma)$ there exists $\epsilon > 0$ such that for all $\mathbf{x_0} \in B_\epsilon(\mathbf{x}^*)$ the sequence $\{\mathbf{x}_k\}$ generated by the Gauss-Newton method is well-defined, converges to $\mathbf{x}^*$ and obeys

$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \frac{c\sigma}{\lambda} \|\mathbf{x}_k - \mathbf{x}^*\| + \frac{c\alpha\gamma}{2\lambda} \|\mathbf{x}_k - \mathbf{x}^*\|^2,$$
$$\|\mathbf{x}_{k+1} - \mathbf{x}^*\| \leq \frac{c\sigma + \lambda}{2\lambda} \|\mathbf{x}_k - \mathbf{x}^*\| < \|\mathbf{x}_k - \mathbf{x}^*\|.$$

Corollary 5.4.1. *Let the assumptions of Theorem 5.4.1 hold. If $\mathbf{F}(\mathbf{x}^*) = 0$, then there exists $\epsilon > 0$ such that for all $\mathbf{x_0} \in B_\epsilon(\mathbf{x}^*)$ the sequence $\{\mathbf{x}_k\}$ generated by the Gauss-Newton method is well-defined and converges quadratically to $\mathbf{x}^*$.*

Theorem 5.4.1 shows that Gauss-Newton method may not be quickly locally convergent and that (when $S(\mathbf{x}^*)$ is too large) it may not be convergent at all. The constant σ plays a crucial role in the convergence. It may be seen as an absolute combined measure of linearity and residual size of the problem because it holds:

$$(J(\mathbf{x}) - J(\mathbf{x}^*))^T \mathbf{F}(\mathbf{x}^*) \simeq S(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}_*),$$

if F is linear or $\mathbf{F}(\mathbf{x}^*) = 0$ then $\sigma = 0$. For the convergence we must look at the ratio $\frac{\sigma}{\lambda}$, which must be less than 1. This can be interpreted as a relative combined measure of linearity and residual size of the problem.

Gauss-Newton method with line-search consists of choosing the Gauss-Newton direction in the line-search algorithm. In this way the method becomes globally convergent and close to $\mathbf{x}^*$ it has quadratic convergence in case $r = 0$.

5.5 Levenberg-Marquardt method

The Levenberg-Marquardt method is a modification of Gauss-Newton method that avoids one of the weaknesses of Gauss-Newton, namely, its behavior when the Jacobian is rank-deficient, or nearly so.

The Levenberg-Marquardt method is derived by modifying the Gauss-Newton model, by adding a regularization term that depends on a strictly positive regularization parameter λ_k:

$$m_k^{LM}(\mathbf{p}) = \frac{1}{2}\|J(\mathbf{x}_k)\mathbf{p} - \mathbf{F}(\mathbf{x}_k)\|^2 + \frac{\lambda_k}{2}\|\mathbf{p}\|^2. \tag{5.2}$$

The minimizer of this model satisfies a modification of the normal equations:

$$(J(\mathbf{x}_k)^T J(\mathbf{x}_k) + \lambda_k I)\mathbf{p} = -J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k).$$

These are indeed the normal equations of the following linear least-squares problem:

$$\min_{\mathbf{p}} \frac{1}{2}\left\| \begin{bmatrix} J(\mathbf{x}_k) \\ \sqrt{\lambda_k} I \end{bmatrix} \mathbf{p} - \begin{bmatrix} \mathbf{F}(\mathbf{x}_k) \\ 0 \end{bmatrix} \right\|^2,$$

which is equivalent to

$$\min_{\mathbf{p}} m_k^{LM}(\mathbf{p}) \tag{5.3}$$

where m_k^{LM} is defined in (5.2). The term that is added ensures that $J(\mathbf{x}_k)^T J(\mathbf{x}_k)$ is positive definite. By using the SVD of $J(\mathbf{x}_k)$, we can write the solution of this problem as

$$\mathbf{x}^* = \sum_{i=1}^{n} \frac{\mathbf{u}_i^T \mathbf{F}}{\sigma_i + \lambda_k} \mathbf{v}_i.$$

Remark that when $\lambda_k \to 0$, $\mathbf{x}^*$ tends to the solution of the Gauss-Newton system. The Levenberg-Marquardt model is then

$$m_k^{LM}(\mathbf{p}) = \frac{1}{2}\|\mathbf{F}(\mathbf{x}_k)\|^2 + J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k)\mathbf{p} + \frac{1}{2}\mathbf{p}^T J(\mathbf{x}_k)^T J(\mathbf{x}_k)\mathbf{p} + \frac{\lambda_k}{2}\|\mathbf{p}\|^2.$$

In the original version of the Levenberg-Marquardt method the parameter λ_k is updated at each iteration, similarly to the trust-region radius in trust-region methods. It is increased or decreased by a certain factor according to whether or not the previous trial step was effective in decreasing f (opposed to the trust-region radius λ_k is decreased if the step is successful). The Levenberg-Marquardt method can indeed be derived from Gauss-Newton method by using a trust-region strategy. Recall that the Gauss-Newton method is like Newton?s method with line search, except that we use the convenient and often effective approximation $J(\mathbf{x})^T J(\mathbf{x})$ for the Hessian. By replacing the line search strategy with a trust-region strategy we obtain the Levenberg-Marquardt method. The second-order Hessian component in (b) is still ignored, however, so the local convergence properties of the two methods are similar.

The following lemma indeed holds.

Lemma 5.5.1. *The solution $\mathbf{p}_k^{LM}$ of the minimization of* (5.2) *is a solution of the trust-region subproblem*

$$\min_{\mathbf{p}} \frac{1}{2}\|J(\mathbf{x}_k)\mathbf{p} - \mathbf{F}(\mathbf{x}_k)\|^2 \text{ subject to } \|\mathbf{p}\| \leq \Delta_k$$

for some $\Delta_k > 0$ if and only if there is a scalar $\lambda_k \geq 0$ such that

$$(J(\mathbf{x}_k)^T J(\mathbf{x}_k) + \lambda_k I)\mathbf{p}_k^{LM} = -J(\mathbf{x}_k)^T \mathbf{F}(\mathbf{x}_k),$$
$$\lambda_k(\Delta_k - \|\mathbf{p}_k^{LM}\|) = 0.$$

This lemma tells us that when the solution $\mathbf{p}_k^{GN}$ of the Gauss-Newton equations (5.1) lies strictly inside the trust region (that is, $\|\mathbf{p}_k^{GN}\| < \Delta_k$), then this step also solves the subproblem (5.3). Otherwise, there is a $\lambda_k > 0$ such that the solution $\mathbf{p}_k^{LM}$ lays on the boundary of the trust-region because as $\lambda_k > 0$ it must hold $\Delta_k = \|\mathbf{p}_k^{LM}\|$.

Chapter 6

Constrained optimization

We are interested in the minimizer $\mathbf{x}^*$ of

$$\begin{array}{rccc} f: & \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ & \mathbf{x} & \longmapsto & f(\mathbf{x}) \end{array}$$

subject to some constraints on the variables, that is we assume that for some

$$\begin{array}{rccc} h: & \mathbb{R}^n & \longrightarrow & \mathbb{R}^p \\ & \mathbf{x} & \longmapsto & \mathbf{h}(\mathbf{x}), \end{array} \qquad \begin{array}{rccc} g: & \mathbb{R}^n & \longrightarrow & \mathbb{R}^m \\ & \mathbf{x} & \longmapsto & \mathbf{g}(\mathbf{x}) \end{array},$$

it holds

$$\mathbf{h}(\mathbf{x}) = \mathbf{0}, \qquad \mathbf{g}(\mathbf{x}) \geq \mathbf{0},^1$$

that is we have p equality constraints and m inequality constraints.
We look for a solution $\mathbf{x}^*$ of the problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \\ &\mathbf{h}(\mathbf{x}) = \mathbf{0}, \\ &\mathbf{g}(\mathbf{x}) \geq \mathbf{0}. \end{aligned}$$

The set

$$\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{h}(\mathbf{x}) = \mathbf{0},\ \mathbf{g}(\mathbf{x}) \geq \mathbf{0}\}$$

is called *feasible set* for the problem. We can then state the problem as

$$\min_{\mathbf{x}\in\Omega} f(\mathbf{x}).$$

We say that an inequality constraint $i \in \{1, \ldots, m\}$ is active in $\mathbf{x}$ if $g_i(\mathbf{x}) = 0$, and inactive if $g_i(\mathbf{x}) > 0$. We denote

$$\mathcal{A}(\mathbf{x}) = \{i \in \{1, \ldots, m\} \mid g_i(\mathbf{x}) = 0\}$$

the set of active constraints in $\mathbf{x}$.

Moreover,

- $\mathbf{x}^* \in \Omega$ is a local minimizer for f if it exists a neighbourhood $\mathcal{N}$ of $\mathbf{x}^*$ such that

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \qquad \forall \mathbf{x} \in \Omega \cap \mathcal{N}.$$

- $\mathbf{x}^*$ is an isolated minimizer for f if it exists a neighbourhood $\mathcal{N}$ of $\mathbf{x}^*$ such that x^* is the only minimizer in $\Omega \cap \mathcal{N}$.

[1] For $\mathbf{v}, \mathbf{w} \in \mathbb{R}^N$, $\mathbf{v} \geq \mathbf{w}$ means $v_i \geq w_i \ \ \forall i = 1, \ldots, N$.

6.1 One equality constraint

Let us consider the problem

$$\min_{\mathbf{x}\in\mathbb{R}^2} x_1 + x_2$$
$$2 - x_1^2 - x_2^2 = 0,$$

i.e., we look for the minimizer of $f(\mathbf{x})$ on the boundary of the circle centred at $(0,0)^T$ and of radius $\sqrt{2}$. From Figure 6.1, in which level curves of $f(\mathbf{x})$ are plotted, it is clear that $\mathbf{x}^* = (-1,-1)^T$. We remark that

$$\nabla f(\mathbf{x}) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \nabla h(\mathbf{x}) = -2 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

Then for each $\mathbf{x}$ on the circumference $\nabla h(\mathbf{x})$ is orthogonal to it and points towards the interior of the circle.
Starting from a point on the circle it is easy to see how to move to remain on the constraint and at the same time to decrease the values of $f(\mathbf{x})$. For example, starting from $\mathbf{x} = (\sqrt{2}, 0)^T$, we can move on the circle clockwise, i.e. following the direction that is tangent to the circle and orthogonal to $\nabla h(\mathbf{x})$ and that is of descent for f in $\mathbf{x}$, i.e. we have to follow the direction $\mathbf{d}$ such that

$$\begin{cases} \nabla h(\mathbf{x})^T \mathbf{d} = 0, \\ \nabla f(\mathbf{x})^T \mathbf{d} < 0. \end{cases}$$

We remark that at the solution, the gradient of the constraint is parallel to the gradient of the function, that is $\exists \mu^* \in \mathbb{R}$ s.t.

$$\nabla f(\mathbf{x}^*) = \mu^* \nabla h(\mathbf{x}^*);$$

in particular it holds $\mu^* = \dfrac{1}{2}$.

Let us assume to have just one equality constraint:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$
$$h(\mathbf{x}) = 0 \quad .$$

If we are at a feasible point $\mathbf{x} : h(\mathbf{x}) = 0$, and we approximate $h(\mathbf{x} + \alpha\mathbf{d})$ with the first order Taylor series, we get:

$$h(\mathbf{x} + \alpha\mathbf{d}) \approx h(\mathbf{x}) + \alpha \nabla h(\mathbf{x})^T \mathbf{d} = \alpha \nabla h(\mathbf{x})^T \mathbf{d},$$

and to decrease $f(\mathbf{x})$ it is necessary that $\mathbf{d}$ is a descent direction for f in $\mathbf{x}$, Then, at first order, we have to move along a direction $\mathbf{d}$ such that

$$\begin{cases} \nabla h(\mathbf{x})^T \mathbf{d} = 0, \\ \nabla f(\mathbf{x})^T \mathbf{d} < 0. \end{cases} \tag{EC}$$

If we are at a feasible point $\mathbf{x} : h(\mathbf{x}) = 0$ and it exists a direction $\mathbf{d}$ that satisfies (EC), we can move along that direction and find a point on the constraint in which f has a lower value, then $\mathbf{x}$ is not $\mathbf{x}^*$. We can then infer the following necessary condition to have that $\mathbf{x}$ is a solution: if $\mathbf{x}$ is a solution, then it cannot exist $\mathbf{d} \in \mathbb{R}^n$ that satisfies (EC).
The only way that $\mathbf{d} \in \mathbb{R}^n$ that satisfies (EC) cannot exist is if $\nabla f(\mathbf{x})$ is parallel to $\nabla h(\mathbf{x})$. The necessary condition becomes then:

$$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \exists \mu^* \in \mathbb{R} \text{ s.t. } \nabla f(\mathbf{x}^*) = \mu^* \nabla h(\mathbf{x}^*).$$

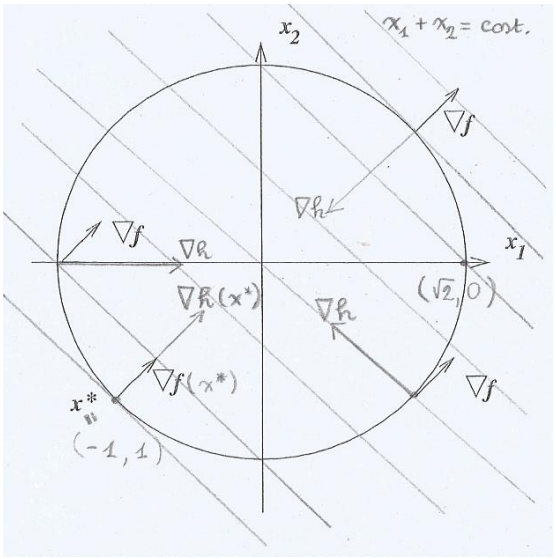

Figure 6.1: Problem of the example, showing constraint and function gradients at various feasible points.

We introduce the Lagrangian function

$$\mathcal{L}(\mathbf{x}, \mu) = f(\mathbf{x}) - \mu h(\mathbf{x}),$$

where μ is called Lagrange multiplier of the equality constraint, and by $\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \mu) = \nabla f(\mathbf{x}) - \mu \nabla h(\mathbf{x})$, the necessary condition becomes

$$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \exists \mu^* \in \mathbb{R} \text{ s.t. } \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}^*, \mu^*) = \mathbf{0}. \tag{NC}$$

This necessary condition is not sufficient. Indeed, in the previous example, if $\widetilde{\mathbf{x}} = (1, 1)^T$, $\exists \widetilde{\mu} \in \mathbb{R}$ s.t. $\nabla f(\widetilde{\mathbf{x}}) = \widetilde{\mu} \nabla h(\widetilde{\mathbf{x}})$, but $\widetilde{\mathbf{x}}$ is not a solution of the problem (this is a maximizer).

6.2 One inequality constraint

Let us consider again the previous problem, in which we replace the equality constraint by an inequality constraint:

$$\min_{\mathbf{x}\in\mathbb{R}^2} x_1 + x_2$$
$$2 - x_1^2 - x_2^2 \geq 0$$

We look for the minimizer of $f(\mathbf{x})$ in the close ball of centre $(0,0)^T$ and radius $\sqrt{2}$. From Figure 6.1 it is clear that the solution is still $\mathbf{x}^* = (-1,-1)^T$, point in which the constraint is active, i.e., $g(\mathbf{x}^*) = 0$. Remark that

$$\nabla f(\mathbf{x}) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \qquad \nabla g(\mathbf{x}) = -2 \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},$$

then for each $\mathbf{x}$ on the circumference of the ball $\nabla g(\mathbf{x})$ is orthogonal to it and points towards the inside.
The gradient of f is parallel to the gradient of g in $\mathbf{x}^*$, that is $\exists \lambda^* \in \mathbb{R}$ s.t.

$$\nabla f(\mathbf{x}^*) = \lambda^* \nabla g(\mathbf{x}^*);$$

in particular it holds $\lambda^* = \dfrac{1}{2}$.

Suppose to have just one inequality constraint:

$$\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x})$$
$$g(\mathbf{x}) \geq 0.$$

If $\mathbf{x} : g(\mathbf{x}) \geq 0$, we have to move along a direction $\mathbf{d}$ remaining on the constraint and at the same time decreasing f; using a first order approximation to g, to remain on the constraint is necessary to have

$$0 \leq g(\mathbf{x} + \alpha\mathbf{d}) \approx g(\mathbf{x}) + \alpha \nabla g(\mathbf{x})^T \mathbf{d},$$

and to decrease $f(\mathbf{x})$ it is necessary that $\mathbf{d}$ is a descent direction for f in $\mathbf{x}$. Then, at first order, we have to move along $\mathbf{d}$ such that

$$\begin{cases} g(\mathbf{x}) + \nabla g(\mathbf{x})^T \mathbf{d} \geq 0 \\ \nabla f(\mathbf{x})^T \mathbf{d} < 0 \end{cases} \tag{IC}$$

If $\mathbf{x} : g(\mathbf{x}) \geq 0$ and it exists a direction $\mathbf{d}$ that satisfies (IC), then we can move along that direction and find a point on the constraint where f has a lower value, then $\mathbf{x}$ is not $\mathbf{x}^*$. We infer then the following necessary condition for $\mathbf{x}$ to be a solution: if $\mathbf{x}$ is a solution, then it does not exist $\mathbf{d} \in \mathbb{R}^n$ that satisfies (IC).

- If the constraint is inactive in $\mathbf{x}$, i.e. $g(\mathbf{x}) > 0$, then the first condition in (IC) is always satisfied, if we choose $\mathbf{d}$ of sufficiently small length, so (IC) becomes

  $$\nabla f(\mathbf{x})^T \mathbf{d} < 0.$$

  It does not exist $\mathbf{d} \in \mathbb{R}^n$ that satisfies this condition if and only if $\nabla f(\mathbf{x}) = \mathbf{0}$. Then the necessary condition becomes

  $$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \nabla f(\mathbf{x}^*) = \mathbf{0}.$$

- If the constraint is inactive in $\mathbf{x}$, that is $g(\mathbf{x}) = 0$, then (IC) becomes

$$\begin{cases} \nabla g(\mathbf{x})^T \mathbf{d} \geq 0 \\ \nabla f(\mathbf{x})^T \mathbf{d} < 0 \end{cases}$$

The first of these two conditions defines a closed semispace of $\mathbb{R}^n$, while the second one an open semispace. It does not exist $\mathbf{d} \in \mathbb{R}^n$ that satisfies the condition if and only if such semispaces have void intersection, i.e., if and only if $\nabla f(\mathbf{x})$ and $\nabla g(\mathbf{x})$ are parallel and point towards the same direction (see Figure 6.2). Then the necessary condition becomes

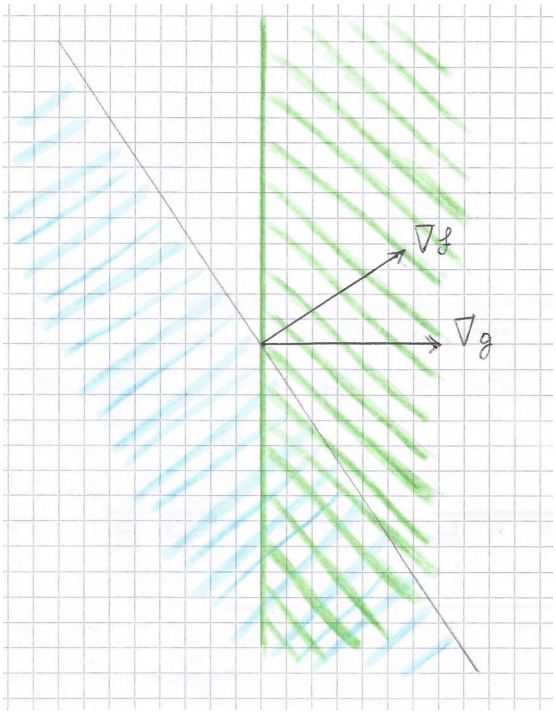

Figure 6.2: The light blue region is the open semispace of the vectors $\mathbf{d}$ such that $\nabla f^T \mathbf{d} < 0$; the green region is the closed semispace of the vectors $\mathbf{d}$ such that $\nabla g^T \mathbf{d} \geq 0$.

$$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \exists \lambda^* > 0 \text{ s.t. } \nabla f(\mathbf{x}^*) = \lambda^* \nabla g(\mathbf{x}^*).$$

The global necessary condition becomes

$$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \exists \lambda^* \geq 0 \text{ s.t. } \nabla f(\mathbf{x}^*) = \lambda^* \nabla g(\mathbf{x}^*), \;\; \lambda^* g(\mathbf{x}^*) = 0.$$

The condition $\lambda^* g(\mathbf{x}^*) = 0$ is called *complementarity condition.*
If the constraint is inactive in $\mathbf{x}^*$, that is $g(\mathbf{x}^*) > 0$, then the complementarity condition requires that $\lambda^* = 0$, and so the necessary condition becomes the one we use in unconstrained optimization.
If in $\mathbf{x}^*$ the constraint is inactive, i.e., $g(\mathbf{x}^*) = 0$, then the complementarity condition does not impose a condition on λ^*.
Introducing the Lagrangian function

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda g(\mathbf{x}),$$

where λ is called Lagrange multiplier of the inequality constraint, and by $\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x},\lambda) = \nabla f(\mathbf{x}) - \lambda\nabla g(\mathbf{x})$, the necessary condition becomes

$$\mathbf{x}^* \text{ is a solution} \quad \Longrightarrow \quad \exists \lambda^* \geq 0 \text{ s.t. } \nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}^*,\lambda^*) = \mathbf{0}, \;\; \lambda^* g(\mathbf{x}^*) = 0.$$

6.3 First order optimality conditions

In general, the Lagrangian function of the problem is defined as

$$\mathcal{L}(\mathbf{x},\boldsymbol{\mu},\boldsymbol{\lambda}) = f(\mathbf{x}) - \sum_{i=1}^{p} \mu_i h_i(\mathbf{x}) - \sum_{i=1}^{m} \lambda_i g_i(\mathbf{x}),$$

where $\boldsymbol{\mu} = (\mu_1,\ldots,\mu_p)^T$ is the vector of Lagrange multipliers for equality constraints and $\boldsymbol{\lambda} = (\lambda_1,\ldots,\lambda_m)^T$ is the vector of Lagrange multipliers for inequality constraints.

Generalizing what we have seen in the examples, if $\mathbf{x} : \mathbf{h}(\mathbf{x}) = \mathbf{0}, \mathbf{g}(\mathbf{x}) \geq \mathbf{0}$, and we want to move along a direction $\mathbf{d}$ remaining on the constraint, at first order we have to move along a direction $\mathbf{d}$ such that

$$\begin{cases} \nabla h_i(\mathbf{x})^T\mathbf{d} = 0 & \forall i = 1,\ldots,p \\ \nabla g_i(\mathbf{x})^T\mathbf{d} \geq 0 & \forall i \in \mathcal{A}(\mathbf{x}). \end{cases}$$

We define

$$F_1(\mathbf{x}) = \{\mathbf{d} \in \mathbb{R}^n \mid \nabla h_i(\mathbf{x})^T\mathbf{d} = 0 \;\; \forall i = 1,\ldots,p, \;\; \nabla g_i(\mathbf{x})^T\mathbf{d} \geq 0 \;\; \forall i \in \mathcal{A}(\mathbf{x})\}$$

set of feasible directions in $\mathbf{x}$. Remark that $F_1(\mathbf{x})$ is a cone[2].
However, this is just a first order approximation (this is exact only if the constraints are linear): in general we have to move along an arc. An arc α parametrized by a parameter $\vartheta \geq 0$ and such that $\boldsymbol{\alpha}(0) = \mathbf{x}$ is said admissible arc in $\mathbf{x}$ if

$$\left\{ \begin{array}{ll} h_i(\boldsymbol{\alpha}(\vartheta)) = 0 & \forall i = 1,\ldots,p \\ g_i(\boldsymbol{\alpha}(\vartheta)) \geq 0 & \forall i \in \mathcal{A}(\mathbf{x}) \end{array} \right\}$$

for ϑ small enough (i.e., for $\vartheta \in [0,\overline{\vartheta}]$ for some $\overline{\vartheta} > 0$). The set

$$T(\mathbf{x}) = \{\mathbf{d} \in \mathbb{R}^n \mid \mathbf{d} = \boldsymbol{\alpha}'(0) \text{ for some } \boldsymbol{\alpha}(\vartheta) \text{ admissible arc in } \mathbf{x}\}$$

of tangent directions to the admissible arcs in $\mathbf{x}$ is a cone and it is called *tangent cone* in $\mathbf{x}$.

Observation 1 $T(\mathbf{x}) \subseteq F_1(\mathbf{x})$.

Proof. Let $\mathbf{d} \in T(\mathbf{x})$. From the definition of tangent cone, $\mathbf{d} = \boldsymbol{\alpha}'(0)$ for some $\boldsymbol{\alpha}(\vartheta)$ admissible arc in $\mathbf{x}$. It holds $\forall i = 1,\ldots,p$

$$0 \underbrace{=}_{h_i(\boldsymbol{\alpha}(\vartheta)) = 0 \;\; \forall \vartheta \in [0,\overline{\vartheta}]} \left[\frac{\mathrm{d}}{\mathrm{d}\vartheta} h_i(\boldsymbol{\alpha}(\vartheta))\right]_{\vartheta=0} = \nabla h_i(\boldsymbol{\alpha}(0))^T\boldsymbol{\alpha}'(0) = \nabla h_i(\mathbf{x})^T\mathbf{d}$$

and $\forall i \in \mathcal{A}(\mathbf{x})$

$$0 \underbrace{\leq}_{\substack{g_i(\boldsymbol{\alpha}(0)) = 0 \text{ and} \\ g_i(\boldsymbol{\alpha}(\vartheta)) \geq 0 \;\; \forall \vartheta \in (0,\overline{\vartheta}]}} \left[\frac{\mathrm{d}}{\mathrm{d}\vartheta} g_i(\boldsymbol{\alpha}(\vartheta))\right]_{\vartheta=0} = \nabla g_i(\boldsymbol{\alpha}(0))^T\boldsymbol{\alpha}'(0) = \nabla g_i(\mathbf{x})^T\mathbf{d}.$$

□

[2] $C \subseteq \mathbb{R}^n$, $C \neq \varnothing$ is a cone if $\forall \mathbf{d} \in C$ it holds $\alpha\mathbf{d} \in C \;\; \forall \alpha \geq 0$

Remark 2 $T(\mathbf{x}) \not\supseteq F_1(\mathbf{x})$.
Let's show this with two examples.

(1) Let us consider the problem

$$\begin{array}{l}\min\limits_{\mathbf{x}\in\mathbb{R}^2} x_1 + x_2 \\ (2 - x_1^2 - x_2^2)^2 = 0\end{array}.$$

Remark that

$$\nabla h(\mathbf{x}) = \begin{pmatrix} 2(2 - x_1^2 - x_2^2)(-2x_1) \\ 2(2 - x_1^2 - x_2^2)(-2x_2) \end{pmatrix} = -4(2 - x_1^2 - x_2^2) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

We notice that $\Omega = \{\mathbf{x} \in \mathbb{R}^2 \mid (2 - x_1^2 - x_2^2)^2 = 0\} = \{\mathbf{x} \in \mathbb{R}^2 \mid 2 - x_1^2 - x_2^2 = 0\}$ is the boundary of the circle of centre $\mathbf{0}$ and radius $\sqrt{2}$. Let $\mathbf{x} \in \Omega$. It is clear that the tangent directions and feasible arcs in $\mathbf{x}$ are just two, then $T(\mathbf{x})$ contains just two elements. However, because $\mathbf{x} \in \Omega$, it holds $2 - x_1^2 - x_2^2 = 0$, and then $\nabla h(\mathbf{x}) = \mathbf{0}$. Then

$$F_1(\mathbf{x}) = \{\mathbf{d} \in \mathbb{R}^2 \mid \underbrace{\nabla h(\mathbf{x})^T}_{=\mathbf{0}} \mathbf{d} = 0\} = \mathbb{R}^2.$$

(2) Let us consider the problem

$$\begin{array}{l}\min\limits_{\mathbf{x}\in\mathbb{R}^2} f(\mathbf{x}) \\ (x_1 - 1)^2 + x_2^2 - 1 = 0, \\ (x_1 + 1)^2 + x_2^2 - 1 = 0.\end{array}$$

Remark that

$$\nabla h_1(\mathbf{x}) = 2 \begin{pmatrix} x_1 - 1 \\ x_2 \end{pmatrix}, \qquad \nabla h_2(\mathbf{x}) = 2 \begin{pmatrix} x_1 + 1 \\ x_2 \end{pmatrix}.$$

The feasible set $\Omega = \{\mathbf{x} \in \mathbb{R}^2 \mid (x_1 - 1)^2 + x_2^2 - 1 = 0, \ \ (x_1 + 1)^2 + x_2^2 - 1 = 0\}$ is the intersection of two circumferences that pass from $\mathbf{0}$ but the first one belongs to the right part of the plane while the second one to the left part of the plane $\mathbb{R}^2$, so $\Omega = \{\mathbf{0}\}$.
There does not exist any admissible arc in $\mathbf{0}$, then there does not exist any tangent direction to an admissible arc in $\mathbf{0}$ and $T(\mathbf{0}) = \varnothing$. On the other hand, because

$$F_1(\mathbf{0}) = \{\mathbf{d} \in \mathbb{R}^2 \mid \nabla h_1(\mathbf{0})^T \mathbf{d} = 0, \ \ \nabla h_2(\mathbf{0})^T \mathbf{d} = 0\} = \{\mathbf{d} \in \mathbb{R}^2 \mid (-2, 0)\mathbf{d} = 0, \ \ (2, 0)\mathbf{d} = 0\},$$

it holds $\mathbf{d} \in F_1(\mathbf{0}) \ \forall \mathbf{d} = (0, d_2) : d_2 \neq 0$; then $F_1(\mathbf{0}) \neq \varnothing$.

Definition 6.3.1. *We say that in* $\mathbf{x}$ *the LICQ (Linear Indipendence Constrains Qualification) holds if the set*

$$\{\nabla h_i(\mathbf{x}) \ \ i = 1, \ldots, p, \qquad \nabla g_i(\mathbf{x}) \ \ i \in \mathcal{A}(\mathbf{x})\}$$

is formed by linearly independent vectors.

Examples In example (1) the LICQ does not hold in $\mathbf{x}$ because $\nabla h(\mathbf{x}) = \mathbf{0}$. In example (2) the LICQ does not hold in $\mathbf{0}$ because $\nabla h_1(\mathbf{0}) \parallel \nabla h_2(\mathbf{0})$.

Lemma 6.3.1. *If LICQ holds in* $\mathbf{x}$*, then* $T(\mathbf{x}) = F_1(\mathbf{x})$.

Example Let us consider the problem

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n} x_1 + x_2\\ &2 - x_1^2 - x_2^2 \geq 0\\ &x_2 \geq 0\end{aligned}$$

We know that

$$\nabla g_1(\mathbf{x}) = -2\begin{pmatrix}x_1\\x_2\end{pmatrix}, \qquad \nabla g_2(\mathbf{x}) = \begin{pmatrix}0\\1\end{pmatrix}.$$

Let $\mathbf{x} = (0, \sqrt{2})^T$. It holds $\mathcal{A}(\mathbf{x}) = \{1\}$, $\nabla g_1(\mathbf{x}) = (0, -2\sqrt{2})^T$. The LICQ holds in $\mathbf{x}$, so

$$\begin{aligned}T(\mathbf{x}) = F_1(\mathbf{x}) &= \{\mathbf{d}\in\mathbb{R}^2 \mid \nabla g_1(\mathbf{x})^T\mathbf{d} \geq 0\} = \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; (0,-2\sqrt{2})\begin{pmatrix}d_1\\d_2\end{pmatrix}\geq 0\right\}\\ &= \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; -2\sqrt{2}d_2 \geq 0\right\} = \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; d_2 \leq 0\right\}.\end{aligned}$$

Let now $\mathbf{x} = (-\sqrt{2}, 0)^T$. It holds $\mathcal{A}(\mathbf{x}) = \{1, 2\}$, $\nabla g_1(\mathbf{x}) = (2\sqrt{2}, 0)^T$, $\nabla g_2(\mathbf{x}) = (0, 1)^T$, then the LICQ holds in $\mathbf{x}$ and

$$\begin{aligned}T(\mathbf{x}) = F_1(\mathbf{x}) &= \{\mathbf{d}\in\mathbb{R}^2 \mid \nabla g_1(\mathbf{x})^T\mathbf{d} \geq 0,\;\; \nabla g_2(\mathbf{x})^T\mathbf{d} \geq 0\}\\ &= \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; (2\sqrt{2},0)\begin{pmatrix}d_1\\d_2\end{pmatrix}\geq 0,\;\; (0,1)\begin{pmatrix}d_1\\d_2\end{pmatrix}\geq 0\right\} = \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; 2\sqrt{2}d_1 \geq 0,\;\; d_2 \geq 0\right\}\\ &= \left\{\begin{pmatrix}d_1\\d_2\end{pmatrix}\in\mathbb{R}^2 \;\middle|\; d_1 \geq 0, d_2 \geq 0\right\}.\end{aligned}$$

Remark 6.3.1. *If in* $\mathbf{x}$ *the LICQ does not hold, we cannot derive a necessary condition of the form (NC). For example we can analyse the problem*

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^2} x_1 + x_2\\ &(2 - x_1^2 - x_2^2)^2 = 0.\end{aligned}$$

For each $\mathbf{x}\in\Omega$*, the LICQ does not hold in* $\mathbf{x}$ *because* $\nabla h(\mathbf{x}) = \mathbf{0}$*. This makes it impossible to proceed as in the case of just one equality constraint to obtain (NC).*

Lemma 6.3.2.

$$\mathbf{x}^* \text{ local minimum point} \quad\Longrightarrow\quad \nexists \mathbf{d}\in T(\mathbf{x}^*) \text{ s.t. } \nabla f(\mathbf{x}^*)^T\mathbf{d} < 0$$

Proof. Let $\mathbf{d}\in T(\mathbf{x}^*)$, i.e., $\mathbf{d} = \boldsymbol{\alpha}'(0)$ for some $\boldsymbol{\alpha}(\vartheta)$ admissible arc in $\mathbf{x}^*$. Because $\boldsymbol{\alpha}(\vartheta)$ is an admissible arc in $\mathbf{x}^*$ and because $\mathbf{x}^*$ is the constrained minimum point, moving from $\mathbf{x}^* = \boldsymbol{\alpha}(0)$ along $\boldsymbol{\alpha}(\vartheta)$ we will find larger values of f:

$$0 \quad\leq\quad \left[\frac{\mathrm{d}f(\boldsymbol{\alpha}(\vartheta))}{\mathrm{d}\vartheta}\right]_{\vartheta=0} \quad=\quad \nabla f(\boldsymbol{\alpha}(0))^T\boldsymbol{\alpha}'(0) \quad=\quad \nabla f(\mathbf{x}^*)^T\mathbf{d}.$$

□

Theorem 6.3.1. *First order necessary condition*

$$\begin{cases}\mathbf{x}^* \text{ is a local minimum point}\\ \text{in } \mathbf{x}^* \text{ LICQ holds}\end{cases} \quad\Longrightarrow\quad \nexists \mathbf{d}\in F_1(\mathbf{x}^*) \text{ s.t. } \nabla f(\mathbf{x}^*)^T\mathbf{d} < 0.$$

The necessary condition is not sufficient. Let's see this with an example. Let us consider the problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^2} x_2 \\ &x_1^2 + x_2 \geq 0. \end{aligned}$$

It holds

$$\Omega = \{\mathbf{x} \in \mathbb{R}^2 \mid x_2 \geq -x_1^2\}.$$

Remark that f is not lower bounded in Ω, then the problem does not admit a solution. Let us consider the point $\mathbf{x}^* = (0,0)^T$. In $\mathbf{x}^*$ the LICQ holds because

$$\nabla g(\mathbf{x}^*) = \Big[\nabla g(\mathbf{x})\Big]_{\mathbf{x}=\mathbf{x}^*} = \left[\begin{pmatrix} 2x_1 \\ 1 \end{pmatrix}\right]_{\mathbf{x}=\mathbf{x}^*} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \neq \mathbf{0}.$$

Because in $\mathbf{x}^*$ the constraint is active, it holds

$$F_1(\mathbf{x}^*) = \{\mathbf{d} \in \mathbb{R}^2 \mid \nabla g(\mathbf{x}^*)^T \mathbf{d} \geq 0\} = \left\{ \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \geq 0 \right\} = \left\{ \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; d_2 \geq 0 \right\}.$$

Remark that $\forall \mathbf{d} \in F_1(\mathbf{x}^*)$

$$\nabla f(\mathbf{x}^*)^T \mathbf{d} = \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = d_2 \geq 0,$$

but $\mathbf{x}^*$ is not a solution.

Lemma 6.3.3. *Farkas Lemma*
Given $C \in \mathbb{R}^{n\times p}, B \in \mathbb{R}^{n\times M}$ we consider the following cone of $\mathbb{R}^n$:

$$K = \{C\mathbf{w} + B\mathbf{y} \mid \mathbf{w} \in \mathbb{R}^p, \mathbf{y} \in \mathbb{R}^M, \mathbf{y} \geq \mathbf{0}\}$$

For each $\mathbf{g} \in \mathbb{R}^n$ exactly one of the following sentences is true:

(a) $\mathbf{g} \in K$

(b) $\exists \mathbf{d} \in \mathbb{R}^n$ *s.t.* $\begin{cases} \mathbf{g}^T\mathbf{d} < 0 & \text{(b.1)} \\ C^T\mathbf{d} = \mathbf{0} & \text{(b.2)} \\ B^T\mathbf{d} \geq \mathbf{0} & \text{(b.3)} \end{cases}$

(a) and (b) are each other opposite: (b) $\iff \neg$(a).

Theorem 6.3.2. *KKT, first order necessary conditions*
If $\mathbf{x}^*$ is a solution in which the LICQ holds, then there exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that

$$\begin{cases} \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) = \mathbf{0}, \\ \mathbf{h}(\mathbf{x}^*) = \mathbf{0}, \\ \mathbf{g}(\mathbf{x}^*) \geq \mathbf{0}, \\ \boldsymbol{\lambda}^* \geq \mathbf{0}, \\ \boldsymbol{\lambda}^{*T} \mathbf{g}(\mathbf{x}^*) = 0. \end{cases}$$

In this case we say that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the Karush-Kuhn-Tucker conditions, usually called KKT conditions. The last condition is the complementarity condition.

Proof. Let $\mathbf{x}^*$ be a solution in which the LICQ holds.
Let

$$\begin{aligned}
&\mathbf{g} = \nabla f(\mathbf{x}^*),\\
&C = \Big(\nabla h_1(\mathbf{x}^*) \mid \cdots \mid \nabla h_p(\mathbf{x}^*)\Big) \in \mathbb{R}^{n\times p},\\
&B = \Big(\nabla g_{i_1}(\mathbf{x}^*) \mid \cdots \mid \nabla g_{i_M}(\mathbf{x}^*)\Big) \in \mathbb{R}^{n\times M}, \qquad \{i_1,\ldots,i_M\} = \mathcal{A}(\mathbf{x}^*)
\end{aligned}$$

and

$$K = \{C\mathbf{w} + B\mathbf{y} \mid \mathbf{w}\in\mathbb{R}^p, \mathbf{y}\in\mathbb{R}^M, \mathbf{y}\geq\mathbf{0}\}. \tag{6.1}$$

With this notation it holds

$$\begin{aligned}
F_1(\mathbf{x}^*) &= \{\mathbf{d}\in\mathbb{R}^n \mid \nabla h_i(\mathbf{x}^*)^T\mathbf{d} = 0 \ \ \forall i = 1,\ldots,p, \ \ \nabla g_i(\mathbf{x}^*)^T\mathbf{d}\geq 0 \ \ \forall i\in\mathcal{A}(\mathbf{x}^*)\}\\
&= \{\mathbf{d}\in\mathbb{R}^n \mid \nabla h_i(\mathbf{x}^*)^T\mathbf{d} = 0 \ \ \forall i = 1,\ldots,p, \ \ \nabla g_{i_j}(\mathbf{x}^*)^T\mathbf{d}\geq 0 \ \ \forall j = 1,\ldots,M\}\\
&= \{\mathbf{d}\in\mathbb{R}^n \mid (C\mathbf{e}_i)^T\mathbf{d} = 0 \ \ \forall i = 1,\ldots,p, \ \ (B\mathbf{e}_j)^T\mathbf{d}\geq 0 \ \ \forall j = 1,\ldots,M\}\\
&= \{\mathbf{d}\in\mathbb{R}^n \mid \mathbf{e}_i^TC^T\mathbf{d} = 0 \ \ \forall i = 1,\ldots,p, \ \ \mathbf{e}_j^TB^T\mathbf{d}\geq 0 \ \ \forall j = 1,\ldots,M\}\\
&= \{\mathbf{d}\in\mathbb{R}^n \mid (C^T\mathbf{d})_i = 0 \ \ \forall i = 1,\ldots,p, \ \ (B^T\mathbf{d})_j\geq 0 \ \ \forall j = 1,\ldots,M\}\\
&= \{\mathbf{d}\in\mathbb{R}^n \mid C^T\mathbf{d} = \mathbf{0}, \ \ B^T\mathbf{d}\geq\mathbf{0}\}.
\end{aligned}$$

By the first order necessary condition, being $\mathbf{x}^*$ a solution in which the LICQ holds,

$$\nexists\mathbf{d}\in F_1(\mathbf{x}^*) \ \text{ s.t. } \ \mathbf{g}^T\mathbf{d} < 0,$$

that is

$$\nexists\mathbf{d}\in\mathbb{R}^n \ \text{ s.t. } \ \begin{cases} \mathbf{g}^T\mathbf{d} < 0\\ C^T\mathbf{d} = \mathbf{0}\\ B^T\mathbf{d}\geq\mathbf{0}\end{cases}$$

Condition (b) of Farkas Lemma does not hold, then (a) must hold and

$$\mathbf{g}\in K.$$

This means that $\nabla f(\mathbf{x}^*)\in K$, that is there exist $\mathbf{w}^*\in\mathbb{R}^p, \mathbf{y}^* = \begin{pmatrix} y^*_{i_1}\\ \vdots\\ y^*_{i_M}\end{pmatrix}\in\mathbb{R}^M, \mathbf{y}^*\geq\mathbf{0}$ such that

$$\begin{aligned}
\nabla f(\mathbf{x}^*) = C\mathbf{w}^* + B\mathbf{y}^* &= \Big(\nabla h_1(\mathbf{x}^*) \mid \cdots \mid \nabla h_p(\mathbf{x}^*)\Big)\mathbf{w}^* + \Big(\nabla g_{i_1}(\mathbf{x}^*) \mid \cdots \mid \nabla g_{i_M}(\mathbf{x}^*)\Big)\mathbf{y}^*\\
&= \sum_{i=1}^p \nabla h_i(\mathbf{x}^*)w_i^* + \sum_{j=1}^M \nabla g_{i_j}(\mathbf{x}^*)y^*_{i_j} = \sum_{i=1}^p w_i^*\nabla h_i(\mathbf{x}^*) + \sum_{i\in\mathcal{A}(\mathbf{x}^*)} y_i^*\nabla g_i(\mathbf{x}^*).
\end{aligned}$$

Set

$$\boldsymbol{\mu}^* = \mathbf{w}^*,$$

$$\boldsymbol{\lambda}^* = \begin{pmatrix}\lambda_1^*\\ \vdots\\ \lambda_m^*\end{pmatrix}, \qquad \lambda_i^* = \begin{cases} y_i^* & \text{if } i\in\mathcal{A}(\mathbf{x}^*)\\ 0 & \text{otherwise}\end{cases}$$

There exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m : \boldsymbol{\lambda}^* \geq \mathbf{0}, \lambda_i^* g_i(\mathbf{x}^*) = 0 \ \ \forall i = 1, \dots, m$ such that

$$\nabla f(\mathbf{x}^*) = \sum_{i=1}^{p} \mu_i^* \nabla h_i(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*),$$

that is, (because $\boldsymbol{\lambda}^* \geq \mathbf{0}$ and $\mathbf{g}(\mathbf{x}^*) \geq \mathbf{0}$) there exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m : \boldsymbol{\lambda}^* \geq \mathbf{0}, \boldsymbol{\lambda}^{*T}\mathbf{g}(\mathbf{x}^*) = 0$ such that

$$\nabla f(\mathbf{x}^*) - \sum_{i=1}^{p} \mu_i^* \nabla h_i(\mathbf{x}^*) - \sum_{i=1}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*) = \mathbf{0}.$$

Because

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \nabla f(\mathbf{x}) - \sum_{i=1}^{p} \mu_i \nabla h_i(\mathbf{x}) - \sum_{i=1}^{m} \lambda_i \nabla g_i(\mathbf{x}),$$

there exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m : \boldsymbol{\lambda}^* \geq \mathbf{0}, \boldsymbol{\lambda}^{*T}\mathbf{g}(\mathbf{x}^*) = 0$ such that

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) = \mathbf{0}.$$

□

Remark 6.3.2. *This necessary condition is not sufficient. Let's see this with an example. We consider again the problem*

$$\begin{aligned} &\min_{\mathbf{x} \in \mathbb{R}^2} x_2 \\ &x_1^2 + x_2 \geq 0. \end{aligned}$$

We have seen that in $\mathbf{x}^* = (0,0)^T$ *the LICQ holds. Because*

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda g(\mathbf{x}) = x_2 - \lambda x_1^2 - \lambda x_2, \\ \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda) = \begin{pmatrix} -2\lambda x_1 \\ 1 - \lambda \end{pmatrix}, \\ \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \lambda) = \begin{pmatrix} 0 \\ 1 - \lambda \end{pmatrix}, \end{aligned}$$

it holds

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \lambda) = \mathbf{0} \quad \iff \quad \lambda = 1,$$

then $(\mathbf{x}^*, \lambda^*)$ *with* $\lambda^* = 1$ *that satisfies the KKT because in* $\mathbf{x}^*$ *the constraint is active. But* $\mathbf{x}^*$ *is not a solution.*

Lemma 6.3.4. *Let* $\mathbf{x}^*$ *be a solution in which the LICQ is satisfied. The first order necessary condition and the KKT conditions are equivalent:*

$\nexists \mathbf{d} \in F_1(\mathbf{x}^*) \quad s.t. \quad \nabla f(\mathbf{x}^*)^T \mathbf{d} < 0 \iff \exists \boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m$ *s.t.* $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ *satisfies the KKT.*

Proof. ($\Longrightarrow$) is the prof of the previous theorem.

($\Longleftarrow$) Because $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies KKT conditions it holds

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \mathbf{0},$$

that is

$$\nabla f(\mathbf{x}^*) = \sum_{i=1}^{p} \mu_i^* \nabla h_i(\mathbf{x}^*) + \sum_{i=1}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*),$$

and then

$$\nabla f(\mathbf{x}^*) = \sum_{i=1}^{p} \mu_i^* \nabla h_i(\mathbf{x}^*) + \sum_{\substack{i=1 \\ i\in\mathcal{A}(\mathbf{x}^*)}}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*) + \sum_{\substack{i=1 \\ i\notin\mathcal{A}(\mathbf{x}^*)}}^{m} \underbrace{\lambda_i^*}_{\substack{=0 \text{ for} \\ \text{complementary}}} \nabla g_i(\mathbf{x}^*)$$

$$= \sum_{i=1}^{p} \mu_i^* \nabla h_i(\mathbf{x}^*) + \sum_{\substack{i=1 \\ i\in\mathcal{A}(\mathbf{x}^*)}}^{m} \underbrace{\lambda_i^*}_{\substack{\geq 0 \\ \text{KKT}}} \nabla g_i(\mathbf{x}^*).$$

So $\nabla f(\mathbf{x}^*)$ belongs to the cone defined in (6.1) and Farkas Lemma guarantees that there do not exist directions $\mathbf{d} \in F_1(\mathbf{x}^*) \quad s.t. \quad \nabla f(\mathbf{x}^*)^T \mathbf{d} < 0$. □

6.4 Second order optimality conditions

Let us assume that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT conditions. Then $\nabla f(\mathbf{x}^*)^T \mathbf{d} \geq 0 \quad \forall \mathbf{d} \in F_1(\mathbf{x}^*)$. If $\nabla f(\mathbf{x}^*)^T \mathbf{d} = 0$, with just first order informations we are not able to establish if along the direction $\mathbf{d}$ the values of f increase or decrease: we need second order informations. Then the directions in the set

$$\{\mathbf{d} \in F_1(\mathbf{x}^*) \mid \nabla f(\mathbf{x}^*)^T \mathbf{d} = 0\}$$

are called critical directions; this set is a cone. If $i \in \mathcal{A}(\mathbf{x}^*)$, i.e., $g_i(\mathbf{x}^*) = 0$, and $\lambda_i^* = 0$ then the i-th inequality constraint is said to be degenerate.
Remark that, given $\mathbf{d} \in F_1(\mathbf{x}^*)$,

$$\begin{aligned}
\nabla f(\mathbf{x}^*)^T \mathbf{d} = 0 \quad & \underbrace{\iff}_{\substack{\text{proof of} \\ \text{prev. thm.}}} \quad \sum_{\substack{i=1 \\ i\in\mathcal{A}(\mathbf{x}^*)}}^{m} \lambda_i^* \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0 \\
\iff \quad & \sum_{\substack{i=1 \\ i\in\mathcal{A}(\mathbf{x}^*) \\ i \text{ degenerate}}}^{m} \underbrace{\lambda_i^*}_{=0} \nabla g_i(\mathbf{x}^*)^T \mathbf{d} \; + \sum_{\substack{i=1 \\ i\in\mathcal{A}(\mathbf{x}^*) \\ i \text{ non degenerate}}}^{m} \underbrace{\lambda_i^*}_{>0} \underbrace{\nabla g_i(\mathbf{x}^*)^T \mathbf{d}}_{\substack{\geq 0 \\ \mathbf{d}\in F_1(\mathbf{x}^*)}} = 0 \\
\iff \quad & \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0 \;\; \forall i \in \mathcal{A}(\mathbf{x}^*) : \; i \text{ non degenerate} \\
\iff \quad & \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0 \;\; \forall i \in \mathcal{A}(\mathbf{x}^*) : \lambda_i^* > 0.
\end{aligned}$$

i.e.,

$$\{\mathbf{d} \in F_1(\mathbf{x}^*) \mid \nabla f(\mathbf{x}^*)^T \mathbf{d} = 0\} \; = \; \{\mathbf{d} \in F_1(\mathbf{x}^*) \mid \nabla g_i(\mathbf{x}^*)^T \mathbf{d} = 0 \;\; \forall i \in \mathcal{A}(\mathbf{x}^*) : \lambda_i^* > 0\}.$$

This last set is the critical cone in $(\mathbf{x}^*, \boldsymbol{\lambda}^*)$ and we denote it with $\mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*)$.

Theorem 6.4.1. *Second order necessary condition*

Let $\mathbf{x}^*$ a solution in which LICQ holds. By the first order necessary condition we know that there exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT. Then

$$\mathbf{d}^T H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)\mathbf{d} \geq 0 \qquad \forall \mathbf{d} \in \mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*),$$

i.e., the matrix

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*) = \begin{pmatrix} \dfrac{\partial \mathcal{L}}{\partial x_1 \partial x_1}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*) & \cdots & \dfrac{\partial \mathcal{L}}{\partial x_1 \partial x_n}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*) \\ \vdots & & \vdots \\ \dfrac{\partial \mathcal{L}}{\partial x_n \partial x_1}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*) & \cdots & \dfrac{\partial \mathcal{L}}{\partial x_n \partial x_n}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*) \end{pmatrix}$$

is the Hessian of $\mathcal{L}(\mathbf{x},\boldsymbol{\mu},\boldsymbol{\lambda})$ with respect to $\mathbf{x}$ in $(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*)$ is semipositive definite with respect to the vectors of the critical cone $\mathcal{C}(\mathbf{x}^*,\boldsymbol{\lambda}^*)$.

Theorem 6.4.2. *Second order sufficient condition*

Let $\mathbf{x}^* \in \Omega$ for which there exist $\boldsymbol{\mu}^* \in \mathbb{R}^p, \boldsymbol{\lambda}^* \in \mathbb{R}^m$ such that $(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*)$ satisfies the KKT. If

$$\mathbf{d}^T H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*)\mathbf{d} > 0 \qquad \forall \mathbf{d} \in \mathcal{C}(\mathbf{x}^*,\boldsymbol{\lambda}^*), \mathbf{d} \neq \mathbf{0},$$

i.e. if $H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*,\boldsymbol{\mu}^*,\boldsymbol{\lambda}^*)$ is positive definite with respect to the vectors of the critical cone $\mathcal{C}(\mathbf{x}^*,\boldsymbol{\lambda}^*)$, then $\mathbf{x}^*$ is a solution.

Remark 6.4.1. *In the unconstrained case, i.e. in the case in which $\Omega = \mathbb{R}^n$, it holds $\mathcal{L} = f$ and the set of admissible directions and the critical cone both are $\mathbb{R}^n$, then these conditions are equivalent to those we have seen in the first part of the course.*

Example 1

Let us consider again problem

$$\begin{array}{l} \min\limits_{\mathbf{x}\in\mathbb{R}^2} x_2 \\ x_1^2 + x_2 \geq 0 \end{array}.$$

We have seen that in $\mathbf{x}^* = (0,0)^T$ the LICQ holds and that $(\mathbf{x}^*,\lambda^*)$ with $\lambda^* = 1$ satisfies the KKT. We have also seen that

$$F_1(\mathbf{x}^*) \quad = \quad \left\{ \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; d_2 \geq 0 \right\},$$

then

$$\begin{aligned} \mathcal{C}(\mathbf{x}^*,\lambda^*) &= \{\mathbf{d} \in F_1(\mathbf{x}^*) \mid \nabla g(\mathbf{x}^*)^T \mathbf{d} = 0\} = \left\{ \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; d_2 \geq 0, \begin{pmatrix} 0 & 1 \end{pmatrix} \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} = 0 \right\} \\ &= \left\{ \begin{pmatrix} d_1 \\ d_2 \end{pmatrix} \in \mathbb{R}^2 \;\middle|\; d_2 = 0 \right\} = \left\{ \begin{pmatrix} d_1 \\ 0 \end{pmatrix} \;\middle|\; d_1 \in \mathbb{R} \right\}. \end{aligned}$$

We have seen that

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x},\lambda) = \begin{pmatrix} -2\lambda x_1 \\ 1 - \lambda \end{pmatrix},$$

then

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x},\lambda) = \begin{pmatrix} -2\lambda & 0 \\ 0 & 0 \end{pmatrix}$$

and

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*,\lambda^*) = \begin{pmatrix} -2 & 0 \\ 0 & 0 \end{pmatrix}.$$

Remark that $\forall \mathbf{d} \in \mathcal{C}(\mathbf{x}^*, \lambda)$, that is for each $\mathbf{d} = \begin{pmatrix} d_1 \\ 0 \end{pmatrix}$ for some $d_1 \in \mathbb{R}$, it holds

$$\mathbf{d}^T H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*, \lambda^*)\mathbf{d} = \begin{pmatrix} d_1 & 0 \end{pmatrix} \begin{pmatrix} -2 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} d_1 \\ 0 \end{pmatrix} = \begin{pmatrix} d_1 & 0 \end{pmatrix} \begin{pmatrix} -2d_1 & 0 \end{pmatrix} = -2d_1^2 < 0,$$

then the second order necessary condition does not hold, and $\mathbf{x}^*$ is not a solution.

Example 2

Let us consider problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^2} -\frac{1}{10}(x_1 - 4)^2 + x_2^2 \\ &x_1^2 + x_2^2 - 1 \geq 0. \end{aligned}$$

After computing

$$\mathcal{L}(\mathbf{x}, \lambda) = f(\mathbf{x}) - \lambda g(\mathbf{x}) = -\frac{1}{10}(x_1 - 4)^2 + x_2^2 - \lambda(x_1^2 + x_2^2 - 1),$$

and

$$\nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda) = \begin{pmatrix} -(1/5)(x_1 - 4) - 2\lambda x_1 \\ 2x_2(1 - \lambda) \end{pmatrix}$$

it is easy to solve the KKT system

$$\begin{cases} \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}, \lambda) = \mathbf{0} \\ g(\mathbf{x}) \geq 0 \\ \lambda \geq 0 \\ \lambda g(\mathbf{x}) = 0 \end{cases}$$

and to see that the solutions are

$$\begin{pmatrix} \mathbf{x}^* \\ \lambda^* \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \dfrac{3}{10} \end{pmatrix}, \qquad \begin{pmatrix} \mathbf{x}^{**} \\ \lambda^{**} \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}$$

and

$$\begin{pmatrix} \mathbf{x}^+ \\ \lambda^+ \end{pmatrix} = \begin{pmatrix} 4/11 \\ \sqrt{105/121} \\ 1 \end{pmatrix}, \qquad \begin{pmatrix} \mathbf{x}^+ \\ \lambda^+ \end{pmatrix} = \begin{pmatrix} 4/11 \\ -\sqrt{105/121} \\ 1 \end{pmatrix}.$$

Let us consider $(\mathbf{x}^*, \lambda^*)$. Because

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x}, \lambda) = \begin{pmatrix} -2\lambda - \dfrac{1}{5} & 0 \\ 0 & 2 - 2\lambda \end{pmatrix},$$

we have that

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*, \lambda^*) = \begin{pmatrix} -\dfrac{4}{5} & 0 \\ 0 & \dfrac{7}{5} \end{pmatrix},$$

which is an indefinite matrix. But $d^T H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*, \lambda^*)d > 0$, $\forall d \in \mathcal{C}(\mathbf{x}^*, \lambda^*) = \{d \in \mathbb{R}^2 s.t. d_1 = 0\}$. Then, for the second order sufficient condition, $\mathbf{x}^*$ is a solution.

On the contrary, in $(\mathbf{x}^{**}, \lambda^{**})$ the constraint is inactive, and as the Hessain matrix of the Lagrangian function in $(\mathbf{x}^{**}, \lambda^{**})$ is indefinite, the point is not a minimum point.

Chapter 7

Optimization methods for Machine Learning

TO DO

Part II

Linear and integer programming

Chapter 8

Linear programming

A linear programming problem (LP) is a constrained minimization problem with $f(\mathbf{x})$ a linear function of variables $x_1, \ldots, x_n$:

$$f(\mathbf{x}) = c_1 x_1 + \ldots + c_n x_n = \mathbf{c}^T \mathbf{x}$$

for some $\mathbf{c} = [c_1, \ldots, c_n]^T \in \mathbb{R}^n$.

Linear programming problems are usually written and analysed in the following form, which is called the *standard form*:

$$\begin{aligned} &\min_{\mathbf{x} \in \mathbb{R}^n} \mathbf{c}^T \mathbf{x} \\ &A\mathbf{x} = \mathbf{b} \\ &\mathbf{x} \geq \mathbf{0} \end{aligned}$$

with $A \in \mathbb{R}^{m \times n}, m < n, \mathrm{rk}(A) = m,\ \mathbf{b} \in \mathbb{R}^m$. The feasible set of the problem is $\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid A\mathbf{x} = \mathbf{b},\ \ \mathbf{x} \geq \mathbf{0}\}$. [1].

We will analyse just the case $m < n$ for the following reasons.

(-) If $m = n$, then $A \in \mathbb{R}^{n \times n}$ with $\mathrm{rk}(A) = n$, then the system $A\mathbf{x} = \mathbf{b}$ has a unique solution $\mathbf{x}$. If $\mathbf{x} \geq \mathbf{0}$ then $\Omega = \{\mathbf{x}\}$ and so the solution to the LP is $\mathbf{x}$, otherwise $\Omega = \varnothing$ and the LP does not have a solution. In every case the problem is trivial.

(-) If $m > n$, then the system $A\mathbf{x} = \mathbf{b}$ has a solution if and only if $\mathbf{b} \in \mathrm{range}(A)$, but it is very unluckily that a vector of $\mathbb{R}^m$ belongs to a subspace of $\mathbb{R}^m$ of dimension n (indeed, if for example $m = 2$ and $n = 1$, it is not luckily that a vector of $\mathbb{R}^2$ belongs to a given line of $\mathbb{R}^2$). It is then highly probable that $\Omega = \varnothing$.

Because

$$\begin{aligned} \mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) &= \mathbf{c}^T \mathbf{x} - \sum_{i=1}^{m} \mu_i (A\mathbf{x} - \mathbf{b})_i - \sum_{i=1}^{n} \lambda_i x_i = \sum_{i=1}^{n} c_i x_i - \sum_{i=1}^{m} \mu_i \big((A\mathbf{x})_i - b_i\big) - \sum_{i=1}^{n} \lambda_i x_i \\ &= \sum_{i=1}^{n} c_i x_i - \sum_{i=1}^{m} \mu_i (a_{i1} x_1 + \ldots + a_{in} x_n - b_i) - \sum_{i=1}^{n} \lambda_i x_i, \end{aligned}$$

[1] While studying constrained minimization problems we have called p the number of equality constraints and m the number of inequality constraints, while in the standard form of LP we have m (independent) equality constraints and n inequality constraints. The difference between these two notations is due to historical reasons: for long time the constrained problems and the LPs have been studied separately, then they have been formulated with different notations.

we have that $\forall j = 1, \dots, n$

$$\Big(\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda})\Big)_j = \frac{\partial}{\partial x_j}\mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = c_j - \sum_{i=1}^{m} \mu_i a_{ij} - \lambda_j = c_j - \sum_{i=1}^{m} (A^T)_{ji}\mu_i - \lambda_j$$
$$= c_j - (A^T\boldsymbol{\mu})_j - \lambda_j = \Big(\mathbf{c} - A^T\boldsymbol{\mu} - \boldsymbol{\lambda}\Big)_j,$$

and so

$$\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda}) = \mathbf{c} - A^T\boldsymbol{\mu} - \boldsymbol{\lambda}.$$

$(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ solves the KKT if and only if

$$\begin{cases} \mathbf{c} - A^T\boldsymbol{\mu}^* - \boldsymbol{\lambda}^* = \mathbf{0}, \\ A\mathbf{x}^* = \mathbf{b}, \\ \mathbf{x}^* \geq \mathbf{0}, \\ \boldsymbol{\lambda}^* \geq \mathbf{0}, \\ \boldsymbol{\lambda}^{*T}\mathbf{x}^* = 0. \end{cases}$$

Because the constraints of the problem are linear, $F_1(x)$ coincide with the tangent cone $T(x)$, $\forall x \in \Omega$.

In the following theorem we will see that the KKT are both necessary and sufficient for an LP.

Theorem 8.0.1. *KKT for LP*

Let $\mathbf{x}^* \in \Omega$. $\mathbf{x}^*$ is a solution to if and only if there exist $\boldsymbol{\mu}^* \in \mathbb{R}^m, \boldsymbol{\lambda}^* \in \mathbb{R}^n$ such that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT.

Proof. We know that ($\Longrightarrow$) is true for every constrained problem, we then need just to prove ($\Longleftarrow$). Remark that

$$f(\mathbf{x}^*) = \mathbf{c}^T\mathbf{x}^* \underbrace{=}_{\text{(KKT 1)}} (A^T\boldsymbol{\mu}^* + \boldsymbol{\lambda}^*)^T\mathbf{x}^* = \boldsymbol{\mu}^{*T}A\mathbf{x}^* + \boldsymbol{\lambda}^{*T}\mathbf{x}^* \underbrace{=}_{\text{(KKT 2,5)}} \boldsymbol{\mu}^{*T}\mathbf{b} \tag{8.1}$$

and that $\forall \mathbf{x} \in \Omega$ holds

$$f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} \underbrace{=}_{\text{(KKT 1)}} (A^T\boldsymbol{\mu}^* + \boldsymbol{\lambda}^*)^T\mathbf{x} = \boldsymbol{\mu}^{*T}A\mathbf{x} + \boldsymbol{\lambda}^{*T}\mathbf{x} \underbrace{\geq}_{\mathbf{x}\in\Omega,\ \text{(KKT 4)}} \boldsymbol{\mu}^{*T}\mathbf{b} = f(\mathbf{x}^*).$$

□

Corollary 8.0.1. *If $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the LP, then*

$$\mathbf{c}^T\mathbf{x}^* = \mathbf{b}^T\boldsymbol{\mu}^*.$$

Proof. This follows from (8.1). □

8.1 How to rewrite an LP in standard form

It always possible to rewrite an LP in standard form by adding new variables to the problem, that are called *slack variables*.

- Example 1. Let us rewrite the following LP in standard form:

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n} \mathbf{c}^T\mathbf{x}\\ &A\mathbf{x}\leq \mathbf{b}\\ &\mathbf{x}\geq \mathbf{0}\end{aligned}$$

Setting

$$\mathbf{s}=\mathbf{b}-A\mathbf{x}\in\mathbb{R}^m,$$

where $s_1,\dots,s_m$ are called slack variables, the problem becomes

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n} \mathbf{c}^T\mathbf{x}\\ &A\mathbf{x}+\mathbf{s}=\mathbf{b}\\ &\mathbf{s}\geq \mathbf{0}\\ &\mathbf{x}\geq \mathbf{0}\end{aligned}$$

Setting

$$\tilde{\mathbf{c}}=\begin{pmatrix}\mathbf{c}\\ \mathbf{0}\end{pmatrix},\qquad \tilde{\mathbf{x}}=\begin{pmatrix}\mathbf{x}\\ \mathbf{s}\end{pmatrix},\qquad \tilde{A}=\begin{pmatrix}A\mid I_m\end{pmatrix} \tag{8.2}$$

the problem becomes

$$\begin{aligned}&\min_{\tilde{\mathbf{x}}\in\mathbb{R}^{n+m}} \tilde{\mathbf{c}}^T\tilde{\mathbf{x}}\\ &\tilde{A}\tilde{\mathbf{x}}=\mathbf{b}\\ &\tilde{\mathbf{x}}\geq \mathbf{0}\end{aligned}$$

that is in standard form.

- Example 2. We consider now the following LP:

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^2} -x_1-2x_2.\\ &-2x_1+x_2\leq 2\\ &-x_1+x_2\leq 3\\ &x_1\leq 3\\ &x_1\geq 0\\ &x_2\geq 0\end{aligned}$$

It is an LP of the form

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^2} \mathbf{c}^T\mathbf{x}\\ &A\mathbf{x}\leq \mathbf{b}\\ &\mathbf{x}\geq \mathbf{0}\end{aligned}$$

with

$$\mathbf{c}=\begin{pmatrix}-1\\ -2\end{pmatrix},\qquad A=\begin{pmatrix}-2 & 1\\ -1 & 1\\ 1 & 0\end{pmatrix},\qquad \mathbf{b}=\begin{pmatrix}2\\ 3\\ 3\end{pmatrix}.$$

We can introduce the slack variables

$$\begin{pmatrix}s_1\\ s_2\\ s_3\end{pmatrix}=\mathbf{s}=\mathbf{b}-A\mathbf{x}=\begin{pmatrix}2\\ 3\\ 3\end{pmatrix}-\begin{pmatrix}-2 & 1\\ -1 & 1\\ 1 & 0\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix}=\begin{pmatrix}2+2x_1-x_2\\ 3+x_1-x_2\\ 3-x_1\end{pmatrix}$$

and by the same setting as in (8.2) the problem becomes

$$\begin{aligned}&\min_{\tilde{\mathbf{x}}\in\mathbb{R}^5} \tilde{\mathbf{c}}^T\tilde{\mathbf{x}}\\ &\tilde{A}\tilde{\mathbf{x}} = \mathbf{b}\\ &\tilde{\mathbf{x}} \geq \mathbf{0}\end{aligned}$$

that is

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^2,\mathbf{s}\in\mathbb{R}^3} -x_1 - 2x_2,\\ &-2x_1 + x_2 + s_1 = 2\\ &-x_1 + x_2 + s_2 = 3\\ &x_1 + s_3 = 3\\ &x_1 \geq 0\\ &x_2 \geq 0\\ &s_1 \geq 0\\ &s_2 \geq 0\\ &s_3 \geq 0.\end{aligned}$$

- Example 3

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n} \mathbf{c}^T\mathbf{x}.\\ &x_2, \ldots, x_n \geq 0\end{aligned}$$

Setting

$$\mathbf{x}' = \begin{pmatrix} x_1' - x_1'' \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

the problem becomes

$$\begin{aligned}&\min_{\mathbf{x}'\in\mathbb{R}^n} \mathbf{c}^T\mathbf{x}'\\ &x_1', x_1'', x_2, \ldots, x_n \geq 0\end{aligned}$$

because, even setting $x_1', x_1'' \geq 0$, the variable $x_1 = x_1' - x_1''$ remains free. Setting

$$\mathbf{c}' = \begin{pmatrix} c_1 \\ -c_1 \\ c_2 \\ \vdots \\ c_n \end{pmatrix}, \qquad \mathbf{x}'' = \begin{pmatrix} x_1' \\ x_1'' \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

the problem becomes

$$\begin{aligned}&\min_{\mathbf{x}''\in\mathbb{R}^{n+1}} \mathbf{c}'^T\mathbf{x}''.\\ &\mathbf{x}'' \geq \mathbf{0}\end{aligned}$$

- Example 4

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n} \mathbf{c}^T\mathbf{x}.\\ &x_1 \geq 3,\\ &x_2, \ldots, x_n \geq 0\end{aligned}$$

Setting $s_1 = x_1 - 3$ and

$$\tilde{\mathbf{c}} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \\ 0 \end{pmatrix}, \qquad \tilde{\mathbf{x}} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \\ s_1 \end{pmatrix}$$

the problem becomes

$$\begin{aligned} &\min_{\tilde{\mathbf{x}} \in \mathbb{R}^{n+1}} \tilde{\mathbf{c}}^T \tilde{\mathbf{x}}. \\ &\tilde{\mathbf{x}} \geq \mathbf{0} \end{aligned}$$

Remark that in each case we have obtained a problem in standard form, but of greater dimension with respect to the original one.

8.2 Primal and dual problems

Problem

$$\begin{aligned} &\min_{\mathbf{x} \in \mathbb{R}^n} \mathbf{c}^T \mathbf{x} \\ &A\mathbf{x} = \mathbf{b} \\ &\mathbf{x} \geq \mathbf{0} \end{aligned}$$

is often called the primal problem. The dual problem is

$$\begin{aligned} &\max_{\boldsymbol{\mu} \in \mathbb{R}^m} \mathbf{b}^T \boldsymbol{\mu} \\ &A^T \boldsymbol{\mu} \leq \mathbf{c}. \end{aligned}$$

Let $\Omega_D = \{\boldsymbol{\mu} \in \mathbb{R}^m \mid A^T \boldsymbol{\mu} \leq \mathbf{c}\}$ the feasible set for the dual problem.

Theorem 8.2.1. *Strong duality for LP*

(i) The primal problem has a solution $\mathbf{x}^*$ if and only if the dual problem has a solution $\boldsymbol{\mu}^*$. In such case the values of the objective functions in the respective solutions coincide:

$$\mathbf{c}^T \mathbf{x}^* = \mathbf{b}^T \boldsymbol{\mu}^*.$$

(ii) If $\mathbf{c}^T \mathbf{x}$ is not lower bounded in Ω, then $\Omega_D = \varnothing$. If $\mathbf{b}^T \boldsymbol{\mu}$ is not upper bounded in Ω_D, then $\Omega = \varnothing$.

Proof. Let us rewrite the dual problem as an LP in the following way:

$$\begin{aligned} &\min_{\boldsymbol{\mu} \in \mathbb{R}^m} -\mathbf{b}^T \boldsymbol{\mu} \\ &\mathbf{c} - A^T \boldsymbol{\mu} \geq \mathbf{0} \end{aligned}$$

Because (denoting by $\mathbf{x}$ the n-dimensional vector of the Lagrange multipliers of the n inequality constraints of the dual problem)

$$\begin{aligned} \mathcal{L}(\boldsymbol{\mu}, \mathbf{x}) &= -\mathbf{b}^T \boldsymbol{\mu} - \sum_{i=1}^{n} x_i (c - A^T \boldsymbol{\mu})_i = -\sum_{i=1}^{m} b_i \mu_i - \sum_{i=1}^{n} x_i (c_i - (A^T \boldsymbol{\mu})_i) \\ &= -\sum_{i=1}^{m} b_i \mu_i - \sum_{i=1}^{n} x_i (c_i - a_{1i}\mu_1 - \ldots - a_{ni}\mu_n), \end{aligned}$$

we have that $\forall j = 1, \dots, m$

$$\Big(\nabla_{\boldsymbol{\mu}} \mathcal{L}(\boldsymbol{\mu}, \mathbf{x})\Big)_j = \frac{\partial}{\partial \mu_j} \mathcal{L}(\boldsymbol{\mu}, \mathbf{x}) = -b_j - \sum_{i=1}^{n} x_i(-a_{ji}) \quad =$$

$$= \quad -b_j + \sum_{i=1}^{n} a_{ji} x_i = -b_j + (A\mathbf{x})_j,$$

and so

$$\nabla_{\boldsymbol{\mu}} \mathcal{L}(\boldsymbol{\mu}, \mathbf{x}) = A\mathbf{x} - \mathbf{b}.$$

Then, a point $(\boldsymbol{\mu}, \mathbf{x})$ satisfies the KKT of the dual problem if and only if

$$\begin{cases} A\mathbf{x} - \mathbf{b} = \mathbf{0} \\ A^T \boldsymbol{\mu} \leq \mathbf{c} \\ \mathbf{x} \geq \mathbf{0} \\ \mathbf{x}^T(\mathbf{c} - A^T \boldsymbol{\mu}) = 0. \end{cases}$$

Setting $\boldsymbol{\lambda} = \mathbf{c} - A^T \boldsymbol{\mu}$, we obtain the KKT of the primal problem, for a point $(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\lambda})$:

$$\begin{cases} A\mathbf{x} - \mathbf{b} = \mathbf{0} \\ A^T \boldsymbol{\mu} + \boldsymbol{\lambda} = \mathbf{c} \\ \boldsymbol{\lambda} \geq \mathbf{0} \\ \mathbf{x} \geq \mathbf{0} \\ \mathbf{x}^T \boldsymbol{\lambda} = 0. \end{cases}$$

Remark that $\boldsymbol{\lambda}$ is a vector of slack variables: the inequality $\mathbf{c} - A^T \boldsymbol{\mu} \geq \mathbf{0}$ becomes $\mathbf{c} - A^T \boldsymbol{\mu} + \boldsymbol{\lambda} = \mathbf{0}$ and $\boldsymbol{\lambda} \geq \mathbf{0}$. It follows that $(\boldsymbol{\mu}^*, \mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the dual if and only if $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the primal. Moreover, being the dual an LP, the KKT of the dual are necessary and sufficient.

(i) Let us assume that the primal problem has a solution $\mathbf{x}^*$. This means that there exist $\boldsymbol{\mu}^* \in \mathbb{R}^m, \boldsymbol{\lambda}^* \in \mathbb{R}^n$ such that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT. So $(\boldsymbol{\mu}^*, \mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the dual, i.e., $\boldsymbol{\mu}^*$ is a solution of the dual problem.
Vice-versa, let us assume that the dual problem has a solution $\boldsymbol{\mu}^*$. This means that there exist $\mathbf{x}^*, \boldsymbol{\lambda}^* \in \mathbb{R}^n$ such that $(\boldsymbol{\mu}^*, \mathbf{x}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the dual. This means that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the primal, so $\mathbf{x}^*$ is solution off the primal problem.
Moreover, in such case, because $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT of the primal, from Corollary 8.0.1, it holds

$$\mathbf{c}^T \mathbf{x}^* = \mathbf{b}^T \boldsymbol{\mu}^*.$$

(ii) We know that if $\mathbf{x}^*$ is solution of the primal and $\boldsymbol{\mu}^*$ is solution of the dual, it holds $\forall \boldsymbol{\mu} \in \Omega_D, \forall \mathbf{x} \in \Omega$

$$\mathbf{b}^T \boldsymbol{\mu} \leq \mathbf{b}^T \boldsymbol{\mu}^* = \mathbf{c}^T \mathbf{x}^* \leq \mathbf{c}^T \mathbf{x}.$$

If $\mathbf{c}^T \mathbf{x}$ is not lower bounded in Ω, than the minimum value of $\mathbf{c}^T \mathbf{x}$ in Ω is $-\infty$, then the maximum value of $\mathbf{b}^T \boldsymbol{\mu}$ in Ω_D is $-\infty$, then $\Omega_D = \varnothing$. Vice-versa, if $\mathbf{b}^T \boldsymbol{\mu}$ is not upper bounded in Ω_D, the maximum value of $\mathbf{b}^T \boldsymbol{\mu}$ in Ω_D is $+\infty$, then the maximum value of $\mathbf{c}^T \mathbf{x}$ in Ω is $+\infty$ and $\Omega = \varnothing$.

□

8.3 Convex and strictly convex problems

The problem

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \\ &\mathbf{h}(\mathbf{x}) = \mathbf{0} \\ &\mathbf{g}(\mathbf{x}) \geq \mathbf{0} \end{aligned}$$

is said convex if both f and Ω are convex; it is strictly convex if f is strictly convex and Ω is convex.

Lemma 8.3.1.

$$\left\{ \begin{array}{l} \forall i = 1,\dots,p \;\; h_i(\mathbf{x}) \textit{ is linear} \\ \forall i = 1,\dots,m \;\; g_i(\mathbf{x}) \textit{ is concave} \end{array} \right. \quad \Longrightarrow \quad \Omega \textit{ is convex.}$$

Proof. Let $\mathbf{x}, \mathbf{y} \in \Omega$ and $t \in [0,1]$. We need to prove that $t\mathbf{x} + (1-t)\mathbf{y} \in \Omega$. We remind that

$$\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid h_i(\mathbf{x}) = 0 \;\; \forall i = 1,\dots,p, \;\; g_i(\mathbf{x}) \geq 0 \;\; \forall i = 1,\dots,m\}.$$

It holds $\forall i = 1,\dots,p$ that

$$h_i(t\mathbf{x} + (1-t)\mathbf{y}) \underbrace{=}_{h_i \text{ is linear}} t\underbrace{h_i(\mathbf{x})}_{=0} + (1-t)\underbrace{h_i(\mathbf{y})}_{=0} = 0$$

and $\forall i = 1,\dots,m$

$$g_i(t\mathbf{x} + (1-t)\mathbf{y}) \underbrace{\geq}_{g_i \text{ is concave}} t\underbrace{g_i(\mathbf{x})}_{\geq 0} + \underbrace{(1-t)}_{\geq 0}\underbrace{g_i(\mathbf{y})}_{\geq 0} \quad \geq \quad 0.$$

□

Corollary 8.3.1.

$$\left\{ \begin{array}{l} f \textit{ is convex} \\ \forall i = 1,\dots,p \;\; h_i(\mathbf{x}) \textit{ is linear} \\ \forall i = 1,\dots,m \;\; g_i(\mathbf{x}) \textit{ is concave} \end{array} \right. \quad \Longrightarrow \quad \textit{the problem is convex.}$$

$$\left\{ \begin{array}{l} f \textit{ is striclty convex} \\ \forall i = 1,\dots,p \;\; h_i(\mathbf{x}) \textit{ is linear} \\ \forall i = 1,\dots,m \;\; g_i(\mathbf{x}) \textit{ is concave} \end{array} \right. \quad \Longrightarrow \quad \textit{the problem is striclty convex.}$$

Remark 8.3.1.

$$\left\{ \begin{array}{l} f \textit{ is striclty convex} \\ \forall i = 1,\dots,p \;\; h_i(\mathbf{x}) \textit{ is linear} \\ \forall i = 1,\dots,m \;\; g_i(\mathbf{x}) \textit{ is concave} \end{array} \right. \quad \Longrightarrow \quad \textit{the KKT are necessary and sufficient.}$$

Proof. We know that for each constrained problem the KKT are necessary, we prove that they are also sufficient. Let $\mathbf{x}^* \in \Omega$ for which there exist $\boldsymbol{\mu}^* \in \mathbb{R}^m, \boldsymbol{\lambda}^* \in \mathbb{R}^n$ such that $(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*)$ satisfies the KKT. Then

$$H_{\mathcal{L},\mathbf{x}}(\mathbf{x}^*, \boldsymbol{\mu}^*, \boldsymbol{\lambda}^*) = H(\mathbf{x}^*) - \sum_{i=1}^{p} \mu_i^* \underbrace{H_{h_i}(\mathbf{x}^*)}_{=0,\, h_i \text{ is linear}} - \sum_{i=1}^{m} \lambda_i^* H_{g_i}(\mathbf{x}^*) \quad =$$

$$= \underbrace{H(\mathbf{x}^*)}_{\substack{\text{posit. def. because} \\ f \text{ strictly convex}}} \underbrace{-\sum_{i=1}^{m} \underbrace{\lambda_i^*}_{\geq 0} \underbrace{H_{g_i}(\mathbf{x}^*)}_{\substack{\text{neg. semidef.} \\ \text{because } g_i \text{ concave}}}}_{\text{neg. semidef.}} = \underbrace{H(\mathbf{x}^*)}_{\text{posit. def.}} + \underbrace{\left(-\sum_{i=1}^{m} \lambda_i^* H_{g_i}(\mathbf{x}^*)\right)}_{\text{posit. semidef.}}$$

is positive definite. Then, for the second order sufficient condition, $\mathbf{x}^*$ is a solution.

($\not\Longleftarrow$) For LPs the KKT sare necessary and sufficient, but $f(\mathbf{x}) = \mathbf{c}^T\mathbf{x}$ is not strictly convex because it is linear (i.e., convex and concave). □

Remark 8.3.2. *LPs are convex (not strictly). Then Ω is convex.*

Proof. f is convex (because it is linear); $\forall i = 1, \ldots, m$

$$h_i(\mathbf{x}) = (A\mathbf{x} - \mathbf{b})_i = (A\mathbf{x})_i - b_i = a_{i1}x_1 + \ldots + a_{in}x_n - b_i$$

is linear. Also $\forall i = 1, \ldots, m$

$$g_i(\mathbf{x}) = x_i$$

is concave (because it is linear). Then, for the corollary, LP is convex. □

8.4 Geometry of Ω

Let us give some preliminary definitions. A *half space* of $\mathbb{R}^n$ is a set of the form

$$\{\mathbf{x} | \mathbf{a}^T\mathbf{x} \geq \mathbf{b}\}, \quad \mathbf{a}, \mathbf{b} \in \mathbb{R}^n, \mathbf{a} \neq \mathbf{0}.$$

Given $\mathbf{x}_1, \ldots, \mathbf{x}_m \in \mathbb{R}^n$, and $\lambda_1, \ldots, \lambda_m \in \mathbb{R}$, $\mathbf{x} = \sum_{i=1}^{m} \lambda_i \mathbf{x}_i$ is a convex combination if $\sum_{i=1}^{m} \lambda_i = 1$ and $\lambda_i \geq 0$ for every i. The *convex hull* conv(X) of a finite set of points X is the set of points which are convex combinations of a finite number of points of X.

A *polyhedron* is an intersection of finitely many half spaces. We say a polyhedron is bounded if it does not contain a line or a half-line. A bounded polyhedron is a *polytope*. A polytope is then the set of solutions of a system of linear equations and linear inequalities. A polytope can also be defined as the convex hull of finitely many points, i.e., it is a set of the form conv(X) for X a finite set. The extreme points of a polytope P are called vertexes and if V is the set of such vertexes it holds $P = \text{conv(V)}$. Then $\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid A\mathbf{x} = \mathbf{b}, \ \ \mathbf{x} \geq \mathbf{0}\}$ is a closed and convex polytope (from Remark 8.3.2).

Definition 8.4.1. *$\mathbf{x} \in \Omega$ is a vertex of Ω if it does not lie on a segment of Ω, i.e., if there do not exist $\mathbf{y}, \mathbf{z} \in \Omega, \mathbf{y}, \mathbf{z} \neq \mathbf{x}$ such that $\mathbf{x} = t\mathbf{y} + (1-t)\mathbf{z}$ for some $t \in (0, 1)$.*

Definition 8.4.2. *$\mathbf{x} \in \mathbb{R}^n$ is a feasible basic point if $\mathbf{x} \in \Omega$ (then $\mathbf{x} \geq \mathbf{0}$) and the columns of A in the set $\{A\mathbf{e}_i \mid i = 1, \ldots, n, \ \ x_i > 0\}$ are linearly independent.*

Notations

Each point $\mathbf{x} \in \Omega$ is such that $\mathbf{x} \geq 0$, then it is possible to reorder its components in a way such that

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix}, \qquad \mathbf{x}_B \in \mathbb{R}^r, \mathbf{x}_B > \mathbf{0}, \ \ \mathbf{x}_N \in \mathbb{R}^{n-r}, \mathbf{x}_N = \mathbf{0}.$$

For the theory we will always consider this partition and the corresponding partition for A:

$$A = \Big(B \mid N\Big), \qquad B \in \mathbb{R}^{m\times r},\ \ N \in \mathbb{R}^{m\times(n-r)}.$$

Remark 8.4.1. *Let* $\mathbf{x} \in \Omega$.

$$\mathbf{x} \textit{ is a feasible basic point} \quad \Longleftrightarrow \quad \begin{array}{l}\textit{the colums of } B\textit{, i.e., } B\mathbf{e}_1, \ldots, B\mathbf{e}_r,\\ \textit{are linearly independent.}\end{array}$$

In such a case, being $B\mathbf{e}_1, \ldots, B\mathbf{e}_r$ *vectors of* $\mathbb{R}^m$*, it holds* $r \le m$ *and* $\mathrm{rk}(B) = r$*. B is called base matrix. If moreover* $r = m$*,* $B \in \mathbb{R}^{m\times m}$ *is invertible.*

Theorem 8.4.1.

$$\mathbf{x} \textit{ is a feasible basic point} \quad \Longleftrightarrow \quad \mathbf{x} \textit{ is a vertex of } \Omega.$$

Proof. Starting from each one of the two assumptions we have $\mathbf{x} \in \Omega$, then we can partition $\mathbf{x}$ and A as explained above.

($\Longrightarrow$) Let us assume by contradiction that $\mathbf{x}$ is not a vertex, i.e. that there exist $\mathbf{y}, \mathbf{z} \in \Omega, \mathbf{y}, \mathbf{z} \neq \mathbf{x}$ such that

$$\mathbf{x} = t\mathbf{y} + (1-t)\mathbf{z} \tag{8.3}$$

for some $t \in (0,1)$. We write

$$\mathbf{y} = \begin{pmatrix}\mathbf{y}_B\\ \mathbf{y}_N\end{pmatrix}, \ \ \mathbf{z} = \begin{pmatrix}\mathbf{z}_B\\ \mathbf{z}_N\end{pmatrix}, \qquad \mathbf{y}_B, \mathbf{z}_B \in \mathbb{R}^r, \ \ \mathbf{y}_N, \mathbf{z}_N \in \mathbb{R}^{n-r}.$$

Remark that

$$\mathbf{0} = \mathbf{x}_N \underbrace{=}_{(8.3)} \underbrace{t}_{>0}\underbrace{\mathbf{y}_N}_{\ge \mathbf{0}} + \underbrace{(1-t)}_{>0}\underbrace{\mathbf{z}_N}_{\ge \mathbf{0}},$$

then $\mathbf{y}_N = \mathbf{z}_N = \mathbf{0}$, i.e.,

$$\mathbf{y}_N = \mathbf{x}_N, \qquad \mathbf{z}_N = \mathbf{x}_N.$$

Because $\mathbf{x} \in \Omega$ it holds

$$\mathbf{b} = A\mathbf{x} = \Big(B \mid N\Big)\begin{pmatrix}\mathbf{x}_B\\ \mathbf{x}_N\end{pmatrix} = B\mathbf{x}_B + N\mathbf{0} = B\mathbf{x}_B,$$

i.e.,

$$B\mathbf{x}_B = \mathbf{b}.$$

Similarly, because $\mathbf{y}, \mathbf{z} \in \Omega$, it holds

$$B\mathbf{y}_B = \mathbf{b}, \qquad B\mathbf{z}_B = \mathbf{b}.$$

Then $\mathbf{y}_B - \mathbf{x}_B \in \ker(B)$. Because $\dim(\ker(B)) = r - \mathrm{rk}(B) = 0$, $\mathbf{y}_B - \mathbf{x}_B = \mathbf{0}$, i.e., $\mathbf{y}_B = \mathbf{x}_B$. Then

$$\mathbf{y} = \begin{pmatrix}\mathbf{y}_B\\ \mathbf{y}_N\end{pmatrix} = \begin{pmatrix}\mathbf{x}_B\\ \mathbf{x}_N\end{pmatrix} = \mathbf{x}.$$

Similarly we can prove that $B(\mathbf{z}_B - \mathbf{x}_B) = \mathbf{0}$ and $\mathbf{z}_B = \mathbf{x}_B$. Then

$$\mathbf{z} = \begin{pmatrix}\mathbf{z}_B\\ \mathbf{z}_N\end{pmatrix} = \begin{pmatrix}\mathbf{x}_B\\ \mathbf{x}_N\end{pmatrix} = \mathbf{x}.$$

We have then found a contradiction with the assumption $\mathbf{y}, \mathbf{z} \neq \mathbf{x}$.

($\Longleftarrow$) We need to prove that the columns of B ($B\mathbf{e}_1, \ldots, B\mathbf{e}_r$), are linearly independent. Let us assume by contradiction that $B\mathbf{e}_1, \ldots, B\mathbf{e}_r$ are not independent, i.e., that it exists $\mathbf{p} = \begin{pmatrix} p_1 \\ \vdots \\ p_r \end{pmatrix} \neq \mathbf{0}$ such that

$$\mathbf{0} = (B\mathbf{e}_1)p_1 + \ldots + (B\mathbf{e}_r)p_r = \begin{pmatrix} B\mathbf{e}_1 \mid \cdots \mid B\mathbf{e}_r \end{pmatrix} \begin{pmatrix} p_1 \\ \vdots \\ p_r \end{pmatrix} = B\mathbf{p}. \qquad (8.3)$$

Because $\mathbf{x}_B > \mathbf{0}$, it exists $\varepsilon > 0$ small enough such that

$$\mathbf{x}_B + \varepsilon\mathbf{p} > \mathbf{0} \qquad \wedge \qquad \mathbf{x}_B - \varepsilon\mathbf{p} > \mathbf{0}.$$

Let

$$\mathbf{y} = \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix}, \quad \mathbf{z} = \begin{pmatrix} \mathbf{x}_B - \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} \quad \in \mathbb{R}^n.$$

Because $\mathbf{p} \neq \mathbf{0}$ it holds $\mathbf{y}, \mathbf{z} \neq \mathbf{x}$. We remark that $\mathbf{y} \in \Omega$ because

$$A\mathbf{y} = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} = B(\mathbf{x}_B + \varepsilon\mathbf{p}) = B\mathbf{x}_B + \varepsilon B\mathbf{p} \quad \underbrace{=}_{(8.3)}$$

$$= \quad B\mathbf{x}_B = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{0} \end{pmatrix} = A\mathbf{x} = \mathbf{b}$$

and

$$\mathbf{y} = \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} \geq \mathbf{0}.$$

Similarly $\mathbf{z} \in \Omega$. Remark that

$$\frac{1}{2}\mathbf{y} + \frac{1}{2}\mathbf{z} = \frac{1}{2}\begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} + \frac{1}{2}\begin{pmatrix} \mathbf{x}_B - \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{0} \end{pmatrix} = \mathbf{x},$$

against the assumption that $\mathbf{x}$ is a vertex of Ω. □

Remark 8.4.2. *The number of vertexes of Ω is less than* $\binom{n}{m} = \dfrac{n!}{m!(n-m)!}$.

Theorem 8.4.2. *Fundamental theorem of linear programming*

(i) If there exist some admissible points, then at least one of them is a feasible basic point.

(ii) If LP has solutions, then at least one of them is a feasible basic point.

Proof. (i) Among all the admissible points we choose the one with the minimum number of positive components. Let k be such number and $\mathbf{x}$ be such point:

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix}, \qquad \mathbf{x}_B \in \mathbb{R}^k, \mathbf{x}_B > \mathbf{0}, \;\; \mathbf{x}_N \in \mathbb{R}^{n-k}, \mathbf{x}_N = \mathbf{0};$$

$$A = \begin{pmatrix} B \mid N \end{pmatrix}, \qquad B \in \mathbb{R}^{m\times k}, \;\; N \in \mathbb{R}^{m\times(n-k)}.$$

Let us assume by contradiction that $\mathbf{x}$ is not a feasible basic point, i.e., because $\mathbf{x}$ is admissible, that $B\mathbf{e}_1, \dots, B\mathbf{e}_k$ are not linearly independent, i.e., that it exists $\mathbf{p} = \begin{pmatrix} p_1 \\ \vdots \\ p_k \end{pmatrix} \neq \mathbf{0}$ such that

$$\mathbf{0} = (B\mathbf{e}_1)p_1 + \ldots + (B\mathbf{e}_k)p_k = \begin{pmatrix} B\mathbf{e}_1 \mid \cdots \mid B\mathbf{e}_k \end{pmatrix} \begin{pmatrix} p_1 \\ \vdots \\ p_k \end{pmatrix} = B\mathbf{p}. \qquad (8.3)$$

Because $\mathbf{x}_B > \mathbf{0}$, it exists $\varepsilon \in \mathbb{R}$ small enough such that

$$\mathbf{x}_B + \varepsilon\mathbf{p} \geq \mathbf{0} \qquad \wedge \qquad \exists i \in \{1, \dots, k\} \text{ s.t. } (\mathbf{x}_B + \varepsilon\mathbf{p})_i = 0.$$

Let

$$\mathbf{y} = \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} \in \mathbb{R}^n.$$

Remark that $\mathbf{y}$ is admissible because

$$A\mathbf{y} = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} = B(\mathbf{x}_B + \varepsilon\mathbf{p}) = B\mathbf{x}_B + \varepsilon B\mathbf{p} \underbrace{=}_{(8.3)}$$

$$= \quad B\mathbf{x}_B = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{0} \end{pmatrix} = A\mathbf{x} = \mathbf{b}$$

and

$$\mathbf{y} = \begin{pmatrix} \mathbf{x}_B + \varepsilon\mathbf{p} \\ \mathbf{0} \end{pmatrix} \geq \mathbf{0}.$$

Then $\mathbf{y}$ is admissible and has at most $k-1$ positive components. This is against the definition of k.

(ii) Among all the solutions of the LP, we choose the one, $\mathbf{x}^*$, with the minimum number of positive components. Let k be such number:

$$\mathbf{x}^* = \begin{pmatrix} \mathbf{x}_B^* \\ \mathbf{x}_N^* \end{pmatrix}, \qquad \mathbf{x}_B^* \in \mathbb{R}^k, \mathbf{x}_B^* > \mathbf{0}, \;\; \mathbf{x}_N^* \in \mathbb{R}^{n-k}, \mathbf{x}_N^* = \mathbf{0};$$
$$A = \begin{pmatrix} B \mid N \end{pmatrix}, \qquad B \in \mathbb{R}^{m \times k}, \;\; N \in \mathbb{R}^{m \times (n-k)}.$$

Let us assume by contradiction that $\mathbf{x}^*$ is not a feasible basic point, i.e., because $\mathbf{x}^*$ is admissible, that $B\mathbf{e}_1, \dots, B\mathbf{e}_k$ are not linearly independent, i.e., it exists $\mathbf{p} = \begin{pmatrix} p_1 \\ \vdots \\ p_k \end{pmatrix} \neq \mathbf{0}$ such that

$$\mathbf{0} = (B\mathbf{e}_1)p_1 + \ldots + (B\mathbf{e}_k)p_k = \begin{pmatrix} B\mathbf{e}_1 \mid \cdots \mid B\mathbf{e}_k \end{pmatrix} \begin{pmatrix} p_1 \\ \vdots \\ p_k \end{pmatrix} = B\mathbf{p}. \qquad (8.3)$$

Because $\mathbf{x}_B^* > \mathbf{0}$, it exists $\overline{\varepsilon} > 0$ such that $\forall \varepsilon \in [0, \overline{\varepsilon}]$

$$\mathbf{x}_B^* + \varepsilon \mathbf{p} \geq \mathbf{0} \qquad \wedge \qquad \mathbf{x}_B^* - \varepsilon \mathbf{p} \geq \mathbf{0}$$

and such that

$$\exists i \in \{1, \ldots, k\} \quad \text{s.t.} \quad (\mathbf{x}_B^* + \overline{\varepsilon}\mathbf{p})_i = 0 \ \ \vee \ \ (\mathbf{x}_B^* - \overline{\varepsilon}\mathbf{p})_i = 0.$$

Let $\forall \varepsilon \geq 0$

$$\mathbf{y}(\varepsilon) = \begin{pmatrix} \mathbf{x}_B^* + \varepsilon \mathbf{p} \\ \mathbf{0} \end{pmatrix}, \quad \mathbf{z}(\varepsilon) = \begin{pmatrix} \mathbf{x}_B^* - \varepsilon \mathbf{p} \\ \mathbf{0} \end{pmatrix} \quad \in \mathbb{R}^n.$$

Remark that $\forall \varepsilon \in [0, \overline{\varepsilon}]$ $\mathbf{y}(\varepsilon)$ is admissible because

$$A\mathbf{y}(\varepsilon) = \Big(B \mid N\Big) \begin{pmatrix} \mathbf{x}_B^* + \varepsilon \mathbf{p} \\ \mathbf{0} \end{pmatrix} = B(\mathbf{x}_B^* + \varepsilon \mathbf{p}) = B\mathbf{x}_B^* + \varepsilon B\mathbf{p} \quad \underbrace{=}_{(8.3)}$$

$$= B\mathbf{x}_B^* = \Big(B \mid N\Big) \begin{pmatrix} \mathbf{x}_B^* \\ \mathbf{0} \end{pmatrix} = A\mathbf{x}^* = \mathbf{b}$$

and

$$\mathbf{y}(\varepsilon) = \begin{pmatrix} \mathbf{x}_B^* + \varepsilon \mathbf{p} \\ \mathbf{0} \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \mathbf{0}.$$

Similarly $\forall \varepsilon \in [0, \overline{\varepsilon}]$ $\mathbf{z}(\varepsilon)$ is admissible.
$\mathbf{x}^*$ is solution of LP, i.e., $f(\mathbf{x}^*) \leq f(\mathbf{x}) \ \ \forall \mathbf{x} \in \Omega$, then $\forall \varepsilon \in [0, \overline{\varepsilon}]$

$$\begin{cases} f(\mathbf{x}^*) \leq f(\mathbf{y}(\varepsilon)) \\ f(\mathbf{x}^*) \leq f(\mathbf{z}(\varepsilon)). \end{cases}$$

Reminding that $\forall \varepsilon \in [0, \overline{\varepsilon}]$, it holds

$$f(\mathbf{y}(\varepsilon)) = \mathbf{c}^T \mathbf{y}(\varepsilon) = \Big(\mathbf{c}_B^T \mid \mathbf{c}_N^T\Big) \begin{pmatrix} \mathbf{x}_B^* + \varepsilon \mathbf{p} \\ \mathbf{0} \end{pmatrix} = \mathbf{c}_B^T(\mathbf{x}_B^* + \varepsilon \mathbf{p}) = \mathbf{c}_B^T \mathbf{x}_B^* + \varepsilon \mathbf{c}_B^T \mathbf{p}$$

$$= \quad \Big(\mathbf{c}_B^T \mid \mathbf{c}_N^T\Big) \begin{pmatrix} \mathbf{x}_B^* \\ \mathbf{0} \end{pmatrix} + \varepsilon \mathbf{c}_B^T \mathbf{p} = \mathbf{c}^T \mathbf{x}^* + \varepsilon \mathbf{c}_B^T \mathbf{p} = f(\mathbf{x}^*) + \varepsilon \mathbf{c}_B^T \mathbf{p} \tag{8.4}$$

and analogously

$$f(\mathbf{z}(\varepsilon)) = f(\mathbf{x}^*) - \varepsilon \mathbf{c}_B^T \mathbf{p}. \tag{8.5}$$

This means that $\forall \varepsilon \in [0, \overline{\varepsilon}]$

$$\begin{cases} f(\mathbf{x}^*) \leq f(\mathbf{x}^*) + \varepsilon \mathbf{c}_B^T \mathbf{p} \\ f(\mathbf{x}^*) \leq f(\mathbf{x}^*) - \varepsilon \mathbf{c}_B^T \mathbf{p}, \end{cases}$$

that is $\forall \varepsilon \in [0, \overline{\varepsilon}]$

$$\begin{cases} \varepsilon \mathbf{c}_B^T \mathbf{p} \geq 0 \\ \varepsilon \mathbf{c}_B^T \mathbf{p} \leq 0, \end{cases}$$

so that

$$\mathbf{c}_B^T \mathbf{p} = 0.$$

Then from (8.4) and (8.5) we have that $\forall \varepsilon \in [0, \overline{\varepsilon}]$

$$f(\mathbf{y}(\varepsilon)) = f(\mathbf{x}^*), \qquad f(\mathbf{z}(\varepsilon)) = f(\mathbf{x}^*).$$

Then $\forall \varepsilon \in [0, \bar{\varepsilon}]$ $\mathbf{y}(\varepsilon)$ and $\mathbf{z}(\varepsilon)$ are solutions of the LP.
We know that

$$\exists i \in \{1, \dots, k\} \quad \text{s.t.} \quad (\mathbf{x}_B^* + \bar{\varepsilon}\mathbf{p})_i = 0 \ \vee \ (\mathbf{x}_B^* - \bar{\varepsilon}\mathbf{p})_i = 0.$$

If $(\mathbf{x}_B^* + \bar{\varepsilon}\mathbf{p})_i = 0$, then

$$\mathbf{y}(\bar{\varepsilon}) = \begin{pmatrix} \mathbf{x}_B^* + \bar{\varepsilon}\mathbf{p} \\ \mathbf{0} \end{pmatrix},$$

even if it is a solution, it has at most $k-1$ positive components; if $(\mathbf{x}_B^* - \bar{\varepsilon}\mathbf{p})_i = 0$, then

$$\mathbf{z}(\bar{\varepsilon}) = \begin{pmatrix} \mathbf{x}_B^* - \bar{\varepsilon}\mathbf{p} \\ \mathbf{0} \end{pmatrix},$$

even if it is a solution, it has at most $k-1$ positive components. In each case we have a contradiction, with the definition of k.

□

8.5 Simplex method

The simplex method is a method that terminates in a finite number of steps that starts from a vertex of Ω and at each steps moves from a vertex to another one. We are going to describe a step of the simplex method under the following assumptions:

(HP1) suppose to have chosen a starting vertex of Ω (we will see how to do that);

(HP2) suppose that the LP is not degenerate, i.e., that each vertex of Ω has exactly m positive components (we will consider the general case later).

Let

$$\mathbf{x}^c = \begin{pmatrix} \mathbf{x}_B^c \\ \mathbf{x}_N^c \end{pmatrix}, \qquad \mathbf{x}_B^c \in \mathbb{R}^m, \mathbf{x}_B^c > \mathbf{0}, \ \ \mathbf{x}_N^c \in \mathbb{R}^{n-m}, \mathbf{x}_N^c = \mathbf{0}$$

be the current vertex. Because $\mathbf{x}^c \in \Omega$,

$$\mathbf{b} = A\mathbf{x}^c = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B^c \\ \mathbf{0} \end{pmatrix} = B\mathbf{x}_B^c; \qquad \mathbf{x}_B^c = B^{-1}\mathbf{b},$$

then

$$\mathbf{x}^c = \begin{pmatrix} B^{-1}\mathbf{b} \\ \mathbf{0} \end{pmatrix}.$$

Remark that

$$f(\mathbf{x}^c) = \mathbf{c}^T\mathbf{x}^c = \begin{pmatrix} \mathbf{c}_B^T \mid \mathbf{c}_N^T \end{pmatrix} \begin{pmatrix} B^{-1}\mathbf{b} \\ \mathbf{0} \end{pmatrix} = \mathbf{c}_B^T B^{-1}\mathbf{b}.$$

We will write a generic $\mathbf{x} \in \Omega$ as

$$\mathbf{x} = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix}, \qquad \mathbf{x}_B \in \mathbb{R}^m, \ \ \mathbf{x}_N \in \mathbb{R}^{n-m}.$$

Remark that $\forall \mathbf{x} \in \Omega$

$$\mathbf{b} = A\mathbf{x} = \begin{pmatrix} B \mid N \end{pmatrix} \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = B\mathbf{x}_B + N\mathbf{x}_N; \qquad B\mathbf{x}_B = \mathbf{b} - N\mathbf{x}_N;$$

$$\mathbf{x}_B = B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N,$$

i.e.,

$$\mathbf{x} = \begin{pmatrix} B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \\ \mathbf{x}_N \end{pmatrix}.$$

Remark that $\forall \mathbf{x} \in \Omega$

$$f(\mathbf{x}) = \mathbf{c}^T\mathbf{x} = \left(\mathbf{c}_B^T \mid \mathbf{c}_N^T\right)\begin{pmatrix} B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \\ \mathbf{x}_N \end{pmatrix} = \mathbf{c}_B^T B^{-1}\mathbf{b} - \mathbf{c}_B^T B^{-1}N\mathbf{x}_N + \mathbf{c}_N^T\mathbf{x}_N$$
$$= \quad f(\mathbf{x}^c) + (\mathbf{c}_N^T - \mathbf{c}_B^T B^{-1}N)\mathbf{x}_N = f(\mathbf{x}^c) + \widehat{\mathbf{c}}_N^T\mathbf{x}_N, \tag{8.6}$$

where we have set

$$\widehat{\mathbf{c}}_N = \mathbf{c}_N - N^T(B^{-1})^T\mathbf{c}_B,$$

that is a vector of $\mathbb{R}^{n-m}$ that does not depend on $\mathbf{x}$, but only on $\mathbf{c}, B, N$. Because of this dependence on B and on N, and because (as we will see) at the end of the step B and N are updated, we have a different $\widehat{\mathbf{c}}_N$ at each step. $\widehat{\mathbf{c}}_N$ is called the *vector of reduced costs*.
If $\widehat{\mathbf{c}}_N \geq \mathbf{0}$, then $\forall \mathbf{x} \in \Omega$

$$f(\mathbf{x}) = f(\mathbf{x}^c) + \underbrace{\widehat{\mathbf{c}}_N^T}_{\geq \mathbf{0}^T}\underbrace{\mathbf{x}_N}_{\geq \mathbf{0}} \quad \geq \quad f(\mathbf{x}^c),$$

i.e., $f(\mathbf{x}^c) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in \Omega$, i.e., the current vertex $\mathbf{x}^c$ is solution of LP. Then, if the optimality test

$$\widehat{\mathbf{c}}_N \geq \mathbf{0}$$

is satisfied, it means we have found a solution of LP: the current vertex $\mathbf{x}^c$.
Otherwise, it means that $\mathbf{x}^c$ is not a solution to LP, then we want to move from $\mathbf{x}^c$ to another vertex of Ω.
If $\mathbf{x} \in \Omega$ is such that $\mathbf{x}_N = \mathbf{0}$, then

$$\mathbf{x} = \begin{pmatrix} B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \\ \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} B^{-1}\mathbf{b} \\ \mathbf{0} \end{pmatrix} = \mathbf{x}^c;$$

then, to obtain a vertex $\mathbf{x}$ different from $\mathbf{x}^c$, it is necessary that $\mathbf{x}_N \neq \mathbf{0}$, i.e. that it exists $i \in \{1, \ldots, n-m\}$ such that $(\mathbf{x}_N)_i > 0$.
As the optimality test is not satisfied, $\widehat{\mathbf{c}}_N < \mathbf{0}$, i.e.,

$$\exists j \in \{1, \ldots, n-m\} \quad \text{s.t.} \quad (\widehat{\mathbf{c}}_N)_j < 0.$$

We will choose as new vertex a point

$$\mathbf{x}^+ = \begin{pmatrix} B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \\ \mathbf{x}_N \end{pmatrix}$$

with

$$\mathbf{x}_N = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ (\mathbf{x}_N)_j \\ 0 \\ \vdots \\ 0 \end{pmatrix} = (\mathbf{x}_N)_j\mathbf{e}_j, \qquad \mathbf{e}_j = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{R}^{n-m},$$

with $(\mathbf{x}_N)_j > 0$.
Remark that

$$f(\mathbf{x}^+) \underbrace{=}_{(8.6)} f(\mathbf{x}^c) + \widehat{\mathbf{c}}_N^T \mathbf{x}_N = f(\mathbf{x}^c) + \begin{pmatrix} (\widehat{\mathbf{c}}_N)_1 & \cdots & (\widehat{\mathbf{c}}_N)_{n-m} \end{pmatrix} \begin{pmatrix} 0 \\ \vdots \\ 0 \\ (\mathbf{x}_N)_j \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

$$= \quad f(\mathbf{x}^c) + (\widehat{\mathbf{c}}_N)_j (\mathbf{x}_N)_j, \tag{8.7}$$

and so

$$\lim_{(\mathbf{x}_N)_j \to +\infty} f(\mathbf{x}^+) = f(\mathbf{x}^c) + \lim_{(\mathbf{x}_N)_j \to +\infty} \underbrace{(\widehat{\mathbf{c}}_N)_j}_{<0} (\mathbf{x}_N)_j = -\infty. \tag{8.8}$$

Remark that $\mathbf{x}^+ \in \Omega$ if and only if

$$\mathbf{0} \leq B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N = B^{-1}\mathbf{b} - (\mathbf{x}_N)_j B^{-1} N \mathbf{e}_j,$$

that is

$$(\mathbf{x}_N)_j (B^{-1}N\mathbf{e}_j)_i \leq (B^{-1}\mathbf{b})_i \qquad \forall i = 1, \ldots, m.$$

Because $B^{-1}N\mathbf{e}_j$ is the j-th column of matrix $B^{-1}N$, it holds

$$(\mathbf{x}_N)_j (B^{-1}N)_{ij} \leq (B^{-1}\mathbf{b})_i \qquad \forall i = 1, \ldots, m,$$

that is

$$\begin{cases} (\mathbf{x}_N)_j \leq \dfrac{(B^{-1}\mathbf{b})_i}{(B^{-1}N)_{ij}} & \forall i : (B^{-1}N)_{ij} > 0 \qquad (1) \\ 0 \leq (B^{-1}\mathbf{b})_i & \forall i : (B^{-1}N)_{ij} = 0 \qquad (2) \\ (\mathbf{x}_N)_j \geq \dfrac{(B^{-1}\mathbf{b})_i}{(B^{-1}N)_{ij}} & \forall i : (B^{-1}N)_{ij} < 0 \qquad (3) \end{cases}$$

Because $B^{-1}\mathbf{b} = \mathbf{x}_B^c \geq \mathbf{0}$, i.e. $(B^{-1}\mathbf{b})_i > 0 \ \ \forall i = 1, \ldots, m$, as the problem is nondegenerate by assumption, (2) is always satisfied and, as $(\mathbf{x}_N)_j > 0$, (3) is always satisfied.
Then, if it exists $i \in \{1, \ldots, m\}$ such that $(B^{-1}N)_{ij} > 0$, then $\mathbf{x}^+ \in \Omega$ if and only if

$$(\mathbf{x}_N)_j \leq \frac{(B^{-1}\mathbf{b})_i}{(B^{-1}N)_{ij}} \qquad \forall i : (B^{-1}N)_{ij} > 0,$$

i.e., said $s \in \{1, \ldots, m\}$ the index such that

$$\frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}} = \min \left\{ \frac{(B^{-1}\mathbf{b})_i}{(B^{-1}N)_{ij}} \;\middle|\; i = 1, \ldots, m, \ (B^{-1}N)_{ij} > 0 \right\},$$

it holds

$$\mathbf{x}^+ \in \Omega \quad \iff \quad (\mathbf{x}_N)_j \leq \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}.$$

Otherwise, if it does not exist $i \in \{1, \ldots, m\}$ such that $(B^{-1}N)_{ij} > 0$, i.e., if $B^{-1}N\mathbf{e}_j \leq \mathbf{0}$, then $\mathbf{x}^+ \in \Omega$ always: $(\mathbf{x}_N)_j$ can be large, and so from (8.8) f is not lower bounded in Ω. Then, if the unboundedness test

$$B^{-1}N\mathbf{e}_j \leq \mathbf{0}$$

is satisfied, the LP does not have a solution.
If on the other hand the test is not satisfied, we choose

$$(\mathbf{x}_N)_j = \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}.$$

Then

$$\mathbf{x}^+ = \begin{pmatrix} \mathbf{x}_B \\ \mathbf{x}_N \end{pmatrix} = \begin{pmatrix} B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N \\ \mathbf{x}_N \end{pmatrix}$$

with

$$\mathbf{x}_N = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}} \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}\mathbf{e}_j$$

and then

$$\mathbf{x}_B = B^{-1}\mathbf{b} - B^{-1}N\mathbf{x}_N = B^{-1}\mathbf{b} - \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}B^{-1}N\mathbf{e}_j,$$

i.e., $\forall i = 1, \ldots, m$

$$(\mathbf{x}_B)_i = (B^{-1}\mathbf{b})_i - \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N\mathbf{e}_j)_i = (B^{-1}\mathbf{b})_i - \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{ij}.$$

Because

$$(\mathbf{x}_B)_s = (B^{-1}\mathbf{b})_s - \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{sj} = 0,$$

it holds

$$\mathbf{x}^+ = \begin{array}{c} \\ \\ \\ s \\ \\ \\ \\ m \\ m+1 \\ \\ \\ m+j \\ \\ \\ \\ n \end{array} \begin{pmatrix} (B^{-1}\mathbf{b})_1 - \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{1j} \\ \vdots \\ (B^{-1}\mathbf{b})_{s-1} - \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{s-1,j} \\ 0 \\ (B^{-1}\mathbf{b})_{s+1} - \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{s+1,j} \\ \vdots \\ (B^{-1}\mathbf{b})_m - \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}(B^{-1}N)_{mj} \\ 0 \\ \vdots \\ 0 \\ \dfrac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}} \\ 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Remark that

$$f(\mathbf{x}^+) \underbrace{=}_{(8.7)} f(\mathbf{x}^c) + (\widehat{\mathbf{c}}_N)_j(\mathbf{x}_N)_j = f(\mathbf{x}^c) + \underbrace{(\widehat{\mathbf{c}}_N)_j}_{<0} \underbrace{\frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}}}_{>0} < f(\mathbf{x}^c).$$

We prove now that $\mathbf{x}^+$ is a vertex of Ω, showing that is a feasible basic point.
Because $\mathbf{x}^+$ is admissible, we have to prove that the columns of A that correspond to the positive elements of $\mathbf{x}^+$ are linearly independent. We have to prove that the vectors

$$A\mathbf{e}_1, \ldots, A\mathbf{e}_{s-1}, A\mathbf{e}_{s+1}, \ldots, A\mathbf{e}_m, A\mathbf{e}_{m+j},$$

are linearly independent, or rather the vectors $B\mathbf{e}_1, \ldots, B\mathbf{e}_{s-1}, B\mathbf{e}_{s+1}, \ldots, B\mathbf{e}_m, N\mathbf{e}_j$, or

$$B\mathbf{e}_1, \ldots, B\mathbf{e}_{s-1}, N\mathbf{e}_j, B\mathbf{e}_{s+1}, \ldots, B\mathbf{e}_m.$$

Theorem 8.5.1. *Let $\mathbf{v}_1, \ldots, \mathbf{v}_m \in \mathbb{R}^m$ be linearly independent. Let $\mathbf{w} = c_1\mathbf{v}_1 + \ldots + c_m\mathbf{v}_m \neq \mathbf{0}$ for some $c_1, \ldots, c_m \in \mathbb{R}$. It exists $s \in \{1, \ldots, m\}$ such that $c_s \neq 0$.*
The vectors $\mathbf{v}_1, \ldots, \mathbf{v}_{s-1}, \mathbf{w}, \mathbf{v}_{s+1}, \ldots, \mathbf{v}_m$ are linearly independent.

Proof. Assume by contradiction that $\mathbf{v}_1, \ldots, \mathbf{v}_{s-1}, \mathbf{w}, \mathbf{v}_{s+1}, \ldots, \mathbf{v}_m$ are linearly dependent, that is, that there exist $d_1, \ldots, d_m \in \mathbb{R}$ not all zero such that

$$d_1\mathbf{v}_1 + \ldots + d_{s-1}\mathbf{v}_{s-1} + d_s\mathbf{w} + d_{s+1}\mathbf{v}_{s+1} + \ldots + d_m\mathbf{v}_m = \mathbf{0}. \qquad (8.3)$$

Remark that $d_s \neq 0$: if it was $d_s = 0$, we would have $d_1\mathbf{v}_1 + \ldots + d_{s-1}\mathbf{v}_{s-1} + d_{s+1}\mathbf{v}_{s+1} + \ldots + d_m\mathbf{v}_m = \mathbf{0}$, then, being by assumption $\mathbf{v}_1, \ldots, \mathbf{v}_{s-1}, \mathbf{v}_{s+1}, \ldots, \mathbf{v}_m$ linearly independent, we would have $d_1 = \ldots = d_m = 0$, against the fact that $d_1, \ldots, d_m$ are not all zero.
For (8.3) it holds

$$\begin{aligned}\mathbf{0} &= d_1\mathbf{v}_1 + \ldots + d_{s-1}\mathbf{v}_{s-1} + d_s(c_1\mathbf{v}_1 + \ldots + c_m\mathbf{v}_m) + d_{s+1}\mathbf{v}_{s+1} + \ldots + d_m\mathbf{v}_m \\ &= (d_1 + d_sc_1)\mathbf{v}_1 + \cdots + (d_{s-1} + d_sc_{s-1})\mathbf{v}_{s-1} + d_sc_s\mathbf{v}_s + (d_{s+1} + d_sc_{s+1})\mathbf{v}_{s+1} + \cdots + (d_m + d_sc_m)\mathbf{v}_m.\end{aligned}$$

Because $d_sc_s \neq 0$, this last is a zero linear combination of $\mathbf{v}_1, \ldots, \mathbf{v}_m$ with coefficients which are not all zero, then $\mathbf{v}_1, \ldots, \mathbf{v}_m$ are linearly dependent, which is a contradiction. □

As the columns of B are linearly independent we have:

$$N\mathbf{e}_j = BB^{-1}N\mathbf{e}_j \underbrace{=}_{\mathbf{q}=B^{-1}N\mathbf{e}_j} B\mathbf{q} = (B\mathbf{e}_1)q_1 + \ldots + (B\mathbf{e}_m)q_m.$$

Moreover the term $q_s = (B^{-1}N\mathbf{e}_j)_s > 0$ from the computation of the index that enters in s. Then from the previous, also

$$B\mathbf{e}_1, \ldots, B\mathbf{e}_{s-1}, N\mathbf{e}_j, B\mathbf{e}_{s+1}, \ldots, B\mathbf{e}_m$$

are linearly independent.
At the end of the step we update B setting

$$B^+ = \Big(B\mathbf{e}_1 \mid \cdots \mid B\mathbf{e}_{s-1} \mid N\mathbf{e}_j \mid B\mathbf{e}_{s+1} \mid \cdots \mid B\mathbf{e}_m\Big),$$

that is invertible, and we update N setting

$$N^+ = \Big(N\mathbf{e}_1 \mid \cdots \mid N\mathbf{e}_{j-1} \mid B\mathbf{e}_s \mid N\mathbf{e}_{j+1} \mid \cdots \mid N\mathbf{e}_{n-m}\Big).$$

The base matrix changes: we say that $B\mathbf{e}_s$ goes out of the basis and $N\mathbf{e}_j$ enters the basis.

The algorithm of a step of simplex method can be sketched in the following way:

0. Given

$$\mathbf{x}^c = \begin{pmatrix} \mathbf{x}_B^c \\ \mathbf{x}_N^c \end{pmatrix}, \qquad \mathbf{x}_B^c \in \mathbb{R}^m, \mathbf{x}_B^c > \mathbf{0}, \;\; \mathbf{x}_N^c \in \mathbb{R}^{n-m}, \mathbf{x}_N^c = \mathbf{0},$$

$$A = \Big(B \mid N\Big), \qquad B \in \mathbb{R}^{m\times m}, B \text{ invertible}, \;\; N \in \mathbb{R}^{m\times(n-r)}.$$

1. Optimality test:
   1. Compute $\mathbf{y} = (B^{-1})^T \mathbf{c}_B$
   2. Compute $\widehat{\mathbf{c}}_N = \mathbf{c}_N - N^T \mathbf{y}$
   3. If $\widehat{\mathbf{c}}_N \geq \mathbf{0}$, then return $\mathbf{x}^c$ as a solution and stop
2. Select $j \in \{1, \ldots, n-m\}$ such that $(\widehat{\mathbf{c}}_N)_j < 0$
3. Compute $\mathbf{q} = B^{-1}(N\mathbf{e}_j)$
4. Unboundedness test:
   If $\mathbf{q} \leq \mathbf{0}$ then return "the problem does not have a solution" and stop
5. Find $s \in \{1, \ldots, m\}$ such that

$$\frac{(\mathbf{x}_B^c)_s}{\mathbf{q}_s} = \min\left\{ \frac{(\mathbf{x}_B^c)_i}{\mathbf{q}_i} \;\middle|\; i = 1, \ldots, m, \;\; \mathbf{q}_i > 0 \right\}$$

   (remember that $B^{-1}\mathbf{b} = \mathbf{x}_B^c$, given in input)
6. Update:
   1. Set

$$\mathbf{x}_N^+ = \left(0, \ldots, 0, \frac{(\mathbf{x}_B^c)_s}{q_s}, 0, \ldots, 0\right)^T, \qquad \mathbf{x}_B^+ = \mathbf{x}_B^c - \frac{(\mathbf{x}_B^c)_s}{q_s}\mathbf{q},$$

   2. Set

$$B^+ = \Big(B\mathbf{e}_1 \mid \cdots \mid B\mathbf{e}_{s-1} \mid N\mathbf{e}_j \mid B\mathbf{e}_{s+1} \mid \cdots \mid B\mathbf{e}_m\Big),$$

$$N^+ = \Big(N\mathbf{e}_1 \mid \cdots \mid N\mathbf{e}_{j-1} \mid B\mathbf{e}_s \mid N\mathbf{e}_{j+1} \mid \cdots \mid N\mathbf{e}_{n-m}\Big).$$

The simplex method terminates in a finite number of steps. Let's prove this. Each time that we move from the current vertex $\mathbf{x}^c$ towards a new vertex $\mathbf{x}^+$, this last one is such that $f(\mathbf{x}^+) < f(\mathbf{x}^c)$, then it is not possible to visit a vertex more than once. Then, because Ω has a finite number of vertexes, and because (from Theorem **??**) at least one of the vertexes of Ω is a solution, in a finite number of steps we will find a vertex that is a solution.

8.5.1 How to choose a starting vertex of Ω

There are various ways to choose a starting vertex in Ω. We will explain one among them. Let us consider the following artificial problem:

$$\begin{aligned} &\min_{\mathbf{x}\in\mathbb{R}^n,\mathbf{z}\in\mathbb{R}^m} \sum_{i=1}^{m} z_i \\ &A\mathbf{x} + E\mathbf{z} = \mathbf{b} \\ &\mathbf{x} \geq \mathbf{0} \\ &\mathbf{z} \geq \mathbf{0} \end{aligned}$$

where A and $\mathbf{b}$ are the data of the original problem, while

$$E = \begin{pmatrix} E_{11} & & \\ & \ddots & \\ & & E_{mm} \end{pmatrix} \in \mathbb{R}^{m\times m}, \qquad E_{ii} = \begin{cases} 1 & \text{se } b_i \geq 0 \\ -1 & \text{se } b_i < 0 \end{cases} \quad \forall i = 1, \dots, m.$$

Set

$$M = \Big(A \mid E\Big), \qquad \mathbf{y} = \begin{pmatrix} \mathbf{x} \\ \mathbf{z} \end{pmatrix},$$

the artificial problem becomes

$$\begin{aligned} &\min_{\mathbf{y}\in\mathbb{R}^{n+m}} \sum_{i=n+1}^{n+m} y_i \\ &M\mathbf{y} = \mathbf{b} \\ &\mathbf{y} \geq \mathbf{0} \end{aligned}$$

The feasible set for this problem is

$$\Omega_a = \{\mathbf{y} \in \mathbb{R}^{n+m} \mid M\mathbf{y} = \mathbf{b},\ \ \mathbf{y} \geq \mathbf{0}\} = \left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{z} \end{pmatrix} \in \mathbb{R}^{n+m} \mid A\mathbf{x} + E\mathbf{z} = \mathbf{b},\ \ \mathbf{x} \geq \mathbf{0},\ \ \mathbf{z} \geq \mathbf{0} \right\}.$$

Let

$$\mathbf{y}_0 = \begin{pmatrix} \mathbf{x}_0 \\ \mathbf{z}_0 \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ |\mathbf{b}| \end{pmatrix}, \qquad |\mathbf{b}| = \begin{pmatrix} |b_1| \\ \vdots \\ |b_m| \end{pmatrix}.$$

Remark that $\mathbf{y}_0 \in \Omega_a$. Indeed

$$M\mathbf{y}_0 = \Big(A \mid E\Big) \begin{pmatrix} \mathbf{0} \\ |\mathbf{b}| \end{pmatrix} = E|\mathbf{b}| = \begin{pmatrix} E_{11} & & \\ & \ddots & \\ & & E_{mm} \end{pmatrix} \begin{pmatrix} |b_1| \\ \vdots \\ |b_m| \end{pmatrix} = \begin{pmatrix} E_{11}|b_1| \\ \vdots \\ E_{mm}|b_m| \end{pmatrix} = (*);$$

because $\forall i = 1, \dots, m$

$$E_{ii}|b_i| = \begin{cases} 1b_i & \text{se } b_i \geq 0 \\ -1(-b_i) & \text{se } b_i < 0 \end{cases} = b_i,$$

we have that

$$(*) = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} = \mathbf{b}.$$

Moreover

$$\mathbf{y}_0 = \begin{pmatrix} \mathbf{0} \\ |\mathbf{b}| \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} = \mathbf{0}.$$

Having proved that $\mathbf{y}_0 \in \Omega_a$, remark that $\mathbf{y}_0$ is a vertex of Ω_a because its positive components correspond to linearly independent columns of M. The positive components of $\mathbf{y}_0$ are between the $(n+1)$-th and the $(n+m)$-th, then they correspond to columns of M between the $(n+1)$-th and the $(n+m)$-th. SO they correspond to columns of $E = \begin{pmatrix} E_{11} & & \\ & \ddots & \\ & & E_{mm} \end{pmatrix}$, which are all linearly independent because $E_{11}, \ldots, E_{mm} \in \{+1, -1\}$.

Remark that function $f(\mathbf{z}) = \sum_{i=1}^{m} z_i$ on the constraint $\mathbf{z} \geq \mathbf{0}$ has the minimum in $\mathbf{z} = \mathbf{0}$. Then, if $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \end{pmatrix} \in \Omega_a$, then $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \end{pmatrix}$ is solution of the artificial problem. It surely exists $\mathbf{x} \in \mathbb{R}^n$ such that $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \end{pmatrix} \in \Omega_a$ because this means that it exists $\mathbf{x} \in \mathbb{R}^n$ such that $A\mathbf{x} = \mathbf{b}$ e $\mathbf{x} \geq \mathbf{0}$. Then it exists $\mathbf{x} \in \Omega$, which is always true. Then the set of the solutions of the artificial problem is

$$\mathcal{S}_a = \left\{ \mathbf{y} = \begin{pmatrix} \mathbf{x} \\ \mathbf{0} \end{pmatrix} \in \mathbb{R}^{n+m} \mid \mathbf{x} \in \Omega \right\}.$$

The simplex method applied to the artificial problem starting from the vertex $\mathbf{y}_0$ will give as a solution a vertex $\mathbf{y}^* = \begin{pmatrix} \mathbf{x}^* \\ \mathbf{0} \end{pmatrix}$ of Ω_a.

Remark that $\mathbf{x}^*$ is a vertex of Ω. Indeed $\mathbf{x}^* \in \Omega$ because $\mathbf{y}^* \in \mathcal{S}_a$. Moreover $\mathbf{x}^*$ is a vertex of Ω: $\mathbf{y}^*$ is a vertex of Ω_a, i.e., the positive components of $\mathbf{y}^*$ correspond to linearly independent columns of M. Because $\mathbf{y}^* = \begin{pmatrix} \mathbf{x}^* \\ \mathbf{0} \end{pmatrix}$, the positive components of $\mathbf{x}^*$ correspond to linearly independent columns of A.

We have then found a vertex of Ω, $\mathbf{x}^*$, from which we can start the simplex method on the original problem.

8.5.2 Generalization of the algorithm to the degenerate case

We have described the algorithm for a step of the simplex method under assumption (HP2) of non degeneracy. If at a step a vertex $\mathbf{x}^c$ is obtained with less then m positive components, at least a component of $\mathbf{x}_B^c$, that is at least a component of $B^{-1}\mathbf{b}$ is zero. If it exists $i \in \{1, \ldots, m\}$ such that $(B^{-1}N)_{ij} > 0$ and $(B^{-1}\mathbf{b})_i = 0$ then

$$\frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}} = \min \left\{ \frac{(B^{-1}\mathbf{b})_i}{(B^{-1}N)_{ij}} \;\middle|\; i = 1, \ldots, m,\ (B^{-1}N)_{ij} > 0 \right\} = 0.$$

If the non boundedness test is not satisfied,

$$(\mathbf{x}_N)_j = \frac{(B^{-1}\mathbf{b})_s}{(B^{-1}N)_{sj}} = 0,$$

and $\mathbf{x}^+ = \mathbf{x}^c$, that is we remain on the same vertex. Then the finite termination of the method is no longer guaranteed.

Bland's rule

Remark that in algorithm **??** steps (2) and (5) may be ambiguous: it may exist more than just one index $j \in \{1, \dots, n-m\}$ such that $(\widehat{\mathbf{c}}_N)_j < 0$ and it may exist more than just one index $s \in \{1, \dots, m\}$ such that $\dfrac{(\mathbf{x}_B^c)_s}{\mathbf{q}_s} = \min\left\{ \dfrac{(\mathbf{x}_B^c)_i}{\mathbf{q}_i} \;\middle|\; i = 1, \dots, m,\ \mathbf{q}_i > 0 \right\}$. In the algorithm we have not specified which index to select in case of ambiguity. Bland's rule requires to choose at step (2) the smallest among the indexes $j \in \{1, \dots, n-m\}$ such that $(\widehat{\mathbf{c}}_N)_j < 0$ and to choose at step (5) the smallest among the indexes $s \in \{1, \dots, m\}$ such that $\dfrac{(\mathbf{x}_B^c)_s}{\mathbf{q}_s} = \min\left\{ \dfrac{(\mathbf{x}_B^c)_i}{\mathbf{q}_i} \;\middle|\; i = 1, \dots, m,\ \mathbf{q}_i > 0 \right\}$. Remark that, as permutations of the components of x are always possible, it is necessary to a-priori enumerate the components of x and the smallest index is referred to such numbering.
It is possible to prove that *the simplex method with Bland's rule always terminates in a finite number of steps, even in the degenerate case.*

Example Let us consider the problem

$$\begin{aligned}
&\min_{\mathbf{x}\in\mathbb{R}^4} -\frac{3}{4}x_1 + 150x_2 - \frac{1}{50}x_3 + 6x_4. \\
&\frac{1}{4}x_1 - 60x_2 - \frac{1}{25}x_3 + 9x_4 \le 0 \\
&\frac{1}{2}x_1 - 90x_2 - \frac{1}{50}x_3 + 3x_4 \le 0 \\
&x_3 \le 1 \\
&\mathbf{x} \ge \mathbf{0}
\end{aligned}$$

The standard form is

$$\begin{aligned}
&\min_{\mathbf{x}\in\mathbb{R}^7} -\frac{3}{4}x_1 - 150x_2 - \frac{1}{50}x_3 + 6x_4, \\
&\frac{1}{4}x_1 - 60x_2 - \frac{1}{25}x_3 + 9x_4 + x_5 = 0 \\
&\frac{1}{2}x_1 - 90x_2 - \frac{1}{50}x_3 + 3x_4 + x_6 = 0 \\
&x_3 + x_7 = 1 \\
&\mathbf{x} \ge \mathbf{0}
\end{aligned}$$

that is

$$\begin{aligned}
&\min_{\mathbf{x}\in\mathbb{R}^7} \mathbf{c}^T\mathbf{x}. \\
&A\mathbf{x} = \mathbf{b} \\
&\mathbf{x} \ge \mathbf{0}
\end{aligned}$$

where

$$\mathbf{c} = \begin{pmatrix} -\frac{3}{4} \\ -150 \\ -\frac{1}{50} \\ 6 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad A = \left(\begin{array}{cccc|ccc} \frac{1}{4} & -60 & -\frac{1}{25} & 9 & 1 & 0 & 0 \\ \frac{1}{2} & -90 & -\frac{1}{50} & 3 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right) = \Big(N \mid B\Big), \qquad \mathbf{b} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

The current point is

$$\mathbf{x}^c = \begin{pmatrix} \mathbf{x}_N^c \\ \mathbf{x}_B^c \end{pmatrix}, \qquad \mathbf{x}_N^c \in \mathbb{R}^4, \mathbf{x}_N^c = \mathbf{0}, \;\; \mathbf{x}_B^c = B^{-1}\mathbf{b} = (I_m)^{-1}\mathbf{b} = \mathbf{b},$$

that is

$$\mathbf{x}^c = \big(0,0,0,0,0,0,1\big)^T.$$

It is a point with less than $m = 3$ positive components, then it is a degenerate point.

If we don't use Bland's rule we can create a loop from which we cannot escape: at first step we take $s = 1$ and $j = 1$, then $s = 2$ and $j = 2$, then $s = 1$ and $j = 3$, then $s = 3$ and $j = 1$, then $s = 1$ and $j = 3$, then $s = 3$ and $j = 1$, and so on: we always remain on the starting vertex.
If we use Bland's rule the process terminates in a finite number of steps.

8.5.3 Advantages and disadvantages of the simplex method

The simplex method has two main advantages.

(V.1) *Finite termination.* We have seen this in the degenerate case but this holds also in general.

(V.2) *A step is cheap.* The only expensive computations in this algorithm are $\mathbf{y} = (B^{-1})^T\mathbf{c}_B$ and $\mathbf{q} = B^{-1}(N\mathbf{e}_j)$. To perform such computations we never compute B^{-1}, but we solve the linear systems $B^T\mathbf{y} = \mathbf{c}_B$ and $B\mathbf{q} = N\mathbf{e}_j$. If an LU factorization of B is available, the first system becomes $(LU)^T\mathbf{y} = \mathbf{c}_B$, or $U^TL^T\mathbf{y} = \mathbf{c}_B$, and the second one $LU\mathbf{q} = N\mathbf{e}_j$. Then we solve the four triangular systems

$$\begin{cases} U^T\mathbf{w} = \mathbf{c}_B \\ L^T\mathbf{y} = \mathbf{w} \end{cases}, \qquad \begin{cases} L\mathbf{w} = N\mathbf{e}_j \\ U\mathbf{q} = \mathbf{w} \end{cases}$$

Each of them cost $O(m)$. The LU factorization of B is not computed ex-novo at each step, which would cost $O(m^3)$: once computed at the beginning of the algorithm, at each step we can compute the factorization by updating the one computed at the previous step. Exploiting the fact that the matrix B at the current step has just a column that is different from that of the previous step, the update of the LU factorization costs $O(m^2)$. Then, a step of the simplex method requires just the solution of four triangular linear systems and the update of the LU factorization of B.

The simplex method also has a disadvantage, to understand it we need the following definition.

Definition 8.5.1. *Complexity*

(-) The complexity of a method that terminates in a finite number of steps is the number of steps performed before the termination.

(-) The complexity of an iterative method is the number of iterations necessary to reach a given accuracy, which translates in a certain stopping criterion $\|\mathbf{x}_k - \mathbf{x}^*\| \leq \text{toll}$.

In both cases, (the tolerance must be fixed for the iterative case) the complexity can either be a linear, or a polynomial, or an exponential... function in the dimension n of the problem. In such cases we will respectively say that the complexity is linear, polynomial, or exponential... in n.

In the worst case, the number of vertexes of Ω is $\dfrac{n!}{m!(n-m)!}$ and we need to visit them all. We then need $\dfrac{n!}{m!(n-m)!}$ steps to terminate the procedure. This is clearly a worst case, which is seldom encountered in practice: the simplex method generally requires only $2m$ or $3m$ steps and works very well. In 1973 Klee and Mint built an example in which Ω is a cube with 2^n vertexes and all of them need to be visited befor reaching the solution. They proved in this way that the simplex method has indeed an exponential complexity in n. People started then looking for methods with polynomial complexity in n. At the end of 1970 Khachiyan proposed the ellipsoid method, which has a polynomial complexity but in practice is slower than the simplex method. In the mid 1980 Karmarkar proposed another method with polynomial complexity, which inspired the interior-point methods, that are widely used nowadays.

Chapter 9

Flow networks problems

In graph theory, a flow network (also known as a transportation network) is a directed graph where each edge has a capacity and each edge receives a flow. The amount of flow on an edge cannot exceed the capacity of the edge. Often in operations research, a directed graph is called a network, the vertices are called nodes and the edges are called arcs. A flow must satisfy the restriction that the amount of flow into a node equals the amount of flow out of it, unless it is a source, which has only outgoing flow, or sink, which has only incoming flow. A network can be used to model traffic in a computer network, circulation with demands, fluids in pipes, currents in an electrical circuit, or anything similar in which something travels through a network of nodes. Flow networks problems can be divided into two large categories: the first is the one of problems for which the passage of the flow through an arc is associated to a cost, such costs are known, and we look for the minimum cost; the second is the one of problems for which the capacities of the edges are known, and we look for the maximum feasible flow.

9.1 Minimum cost flow problem

Let us consider a network, represented by an oriented graph

$$G = (V, E), \qquad V = \{1, \dots, n\}, \qquad E = \{e_1, \dots, e_m\} \subseteq V \times V.$$

In a minimum cost flow problem, each edge $(i, j) \in E$ has a given cost c_{ij}, and the cost of sending a part of the flow x_{ij} across the edge is $c_{ij}x_{ij}$. The objective is to send a given amount of flow from the source to the sink, at the lowest possible price. These problems can be formulated in this way.

For each $i, j = 1, \dots, n : (i, j) \in E$ we denote then with $x_{ij} \geq 0$ the part of the flow that passes from i to j and these are going to be the variables of our problem. The total cost that we want to minimize is $\sum_{\substack{i=1 \\ (i,j)\in E}}^{n} c_{ij}x_{ij}$.

We formulate the problem as follows:

$$\begin{aligned}
&\min_{x_{ij}} \sum_{\substack{i,j=1\\(i,j)\in E}}^{n} c_{ij}x_{ij},\\
&\sum_{\substack{j=1\\(i,j)\in E}}^{n} x_{ij} - \sum_{\substack{j=1\\(j,i)\in E}}^{n} x_{ji} = b_i \qquad \forall i = 1,\dots,n\\
&x_{ij} \geq 0 \qquad \forall i,j = 1,\dots,n : (i,j) \in E
\end{aligned}$$

where $b_1,\dots,b_n$ are assigned quantities and $\sum_{\substack{j=1\\(i,j)\in E}}^{n} x_{ij}$ is the outgoing flow from node i, $\sum_{\substack{j=1\\(j,i)\in E}}^{n} x_{ji}$ is the ingoing flow in node i.

- If $b_i = 0$, i is said a transit node (the amount of flow that enters in the node i is the same as that which goes out);
- If $b_i > 0$, i is said a supply node (the flow that goes out from i is larger than that which goes in);
- If $b_i < 0$, i is said a termination node (the flow that enters in i is larger than that which goes out).

This is a LP because the objective function is linear. We can put it in standard form. We define

$$\mathbf{c} = \begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix}, \qquad \mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix},$$

where $\forall k = 1,\dots,m$, c_k is the cost associated to the passage of the flow along the edge e_k, and x_k is the flow (to be found) that passes along the edge e_k. We define the incidence matrix node-edge of G as the matrix $A \in \mathbb{R}^{n\times m}$ such that

$$A_{ik} = \begin{cases} 1 & \text{if } e_k = (i,j) \text{ for some } j = 1,\dots,n\\ -1 & \text{if } e_k = (j,i) \text{ for some } j = 1,\dots,n\\ 0 & \text{altrimenti} \end{cases}$$

With this notations the total cost (i.e., the objective function) is $\sum_{k=1}^{m} c_k x_k = \mathbf{c}^T\mathbf{x}$, the second constraint becomes $\mathbf{x} \geq \mathbf{0}$ and the first one $A\mathbf{x} = \mathbf{b}$. Indeed $A\mathbf{x} = \mathbf{b}$ means $(A\mathbf{x})_i = b_i \ \ \forall i = 1,\dots,n$, where

$$(A\mathbf{x})_i = \sum_{k=1}^{m} A_{ik}x_k = \underbrace{\sum_{\substack{k=1\\e_k=(i,*)}}^{m} x_k}_{\substack{\text{outgoing}\\\text{flow from } i}} - \underbrace{\sum_{\substack{k=1\\e_k=(*,i)}}^{m} x_k}_{\substack{\text{ingoing flow}\\\text{in } i}}.$$

We can then rewrite the LP as

$$\begin{aligned}
&\min_{\mathbf{x}\in\mathbb{R}^m} \mathbf{c}^T\mathbf{x}.\\
&A\mathbf{x} = \mathbf{b}\\
&\mathbf{x} \geq \mathbf{0}
\end{aligned}$$

To solve the LP we could use the simplex algorithm, but usually the DIJKSTRA algorithm is preferred that, exploiting the special structure of matrix A, has a quadratic complexity (and so polynomial) in m, and not exponential as the simplex.

9.2 Maximum flow problem

We consider a network, represented by an oriented graph

$$G = (V, E), \qquad V = \{1, \dots, n\}, \qquad E = \{e_1, \dots, e_m\} \subseteq V \times V,$$

with a source s and a sink t.

For each $i, j = 1, \dots, n : (i, j) \in E$ we know the capacity $c_{ij} \in \mathbb{Z}_{\geq 0}$ of the edge (i, j), i.e., the maximum amount of flow that can pass through an edge (i, j).
A feasible flow of G is a vector $\mathbf{f} = \left(f_{ij}\right)_{(i,j)\in E} \in \mathbb{R}^m$, where each $f_{ij} \in \mathbb{Z}_{\geq 0}$ represents the flow from i to j, such that

$$\begin{cases} \overbrace{\sum_{\substack{j=1 \\ (i,j)\in E}}^{n} f_{ij}}^{\text{outgoing flow from } i} - \overbrace{\sum_{\substack{j=1 \\ (j,i)\in E}}^{n} f_{ji}}^{\text{ingoing flow in } i} = 0 \qquad \forall i = 2, \dots, n-1 \\ 0 \leq f_{ij} \leq c_{ij} \qquad \forall i, j = 1, \dots, n : (i, j) \in E \end{cases}$$

Given a feasible flow $\mathbf{f}$, we call value of the flow the amount of flow passing from the source to the sink:

$$v = \sum_{\substack{j=1 \\ (s,j)\in E}}^{n} f_{sj}.$$

We want to find the maximum value of a feasible flow, i.e., we have to solve the problem

$$\begin{aligned} &\max_{v\in\mathbb{R}} v, \\ &\overbrace{\sum_{\substack{j=1 \\ (s,j)\in E}}^{n} f_{sj}}^{\text{outgoing flow from } s} = v \\ &\overbrace{\sum_{\substack{j=1 \\ (i,j)\in E}}^{n} f_{ij}}^{\text{outgoing flow from } i} - \overbrace{\sum_{\substack{j=1 \\ (j,i)\in E}}^{n} f_{ji}}^{\text{ingoing flow in } i} = 0 \qquad \forall i = 2, \dots, n-1 \\ &\overbrace{\sum_{\substack{j=1 \\ (j,t)\in E}}^{n} f_{jt}}^{\text{ingoing flow in } t} = v \\ &0 \leq f_{ij} \leq c_{ij} \qquad \forall i, j = 1, \dots, n : (i, j) \in E \end{aligned}$$

This is an LP:

$$
\begin{aligned}
&\min_{v\in\mathbb{R}} -v. \\
&\overbrace{\sum_{\substack{j=1\\(s,j)\in E}}^{n} f_{sj}}^{\text{outgoing flow from } s} = v \\
&\overbrace{\sum_{\substack{j=1\\(i,j)\in E}}^{n} f_{ij}}^{\text{outgoing flow from } i} - \overbrace{\sum_{\substack{j=1\\(j,i)\in E}}^{n} f_{ji}}^{\text{ingoing flow in } i} = 0 \qquad \forall i = 2,\dots,n-1 \\
&\overbrace{\sum_{\substack{j=1\\(j,t)\in E}}^{n} f_{jt}}^{\text{ingoing flow in } t} = v \\
&0 \le f_{ij} \le c_{ij} \qquad \forall i,j = 1,\dots,n : (i,j) \in E
\end{aligned}
$$

We can put this in standard form. Let's define

$$
\mathbf{c} = \begin{pmatrix} c_1 \\ \vdots \\ c_m \end{pmatrix}, \qquad \mathbf{f} = \begin{pmatrix} f_1 \\ \vdots \\ f_m \end{pmatrix},
$$

where $\forall k = 1,\dots,m \quad c_k$ is the capacity of the edge e_k, and f_k is the amount of flow (to be found) that passes through the edge e_k. We denote with A the incidence matrix of G. Let

$$
\mathbf{b} = v(\mathbf{e}_1 - \mathbf{e}_n) = \begin{pmatrix} v \\ 0 \\ \vdots \\ 0 \\ -v \end{pmatrix}.
$$

With these notations the last constraint becomes $\mathbf{0} \le \mathbf{f} \le \mathbf{c}$ and the other constraints become $A\mathbf{f} = \mathbf{b}$. Indeed $A\mathbf{f} = \mathbf{b}$ means $(A\mathbf{f})_i = b_i \ \ \forall i = 1,\dots,n$, where

$$
(A\mathbf{f})_i = \sum_{k=1}^{m} A_{ik} f_k = \underbrace{\sum_{\substack{k=1\\e_k=(i,*)}}^{m} f_k}_{\text{outgoing flow from } i} - \underbrace{\sum_{\substack{k=1\\e_k=(*,i)}}^{m} f_k}_{\text{ingoing flow in } i}.
$$

Then we can rewrite the LP as

$$
\begin{aligned}
&\min_{v\in\mathbb{R}} -v. \\
&A\mathbf{f} = \mathbf{b} \\
&\mathbf{0} \le \mathbf{f} \le \mathbf{c}
\end{aligned}
$$

The admissible set of the LP is

$$
\Omega = \{\mathbf{f} \in \mathbb{R}^m \mid A\mathbf{f} = \mathbf{b}, \ \ \mathbf{0} \le \mathbf{f} \le \mathbf{c}\}.
$$

A LP with $\Omega \neq \varnothing$ and with lower bounded objective function in Ω has a solution. Our LP is such that $\Omega \neq \varnothing$, indeed the zero flow $\mathbf{f} = \mathbf{0}$ belongs to Ω as it corresponds to $\mathbf{b} = v(\mathbf{e}_1 - \mathbf{e}_n) = 0(\mathbf{e}_1 - \mathbf{e}_n) = \mathbf{0}$. Moreover the objective function is lower bounded in Ω because, as

$$v = \sum_{\substack{j=1 \\ (s,j)\in E}}^{n} f_{sj} \leq \sum_{\substack{j=1 \\ (s,j)\in E}}^{n} c_{sj},$$

it holds

$$-v \geq -\sum_{\substack{j=1 \\ (s,j)\in E}}^{n} c_{sj}.$$

Then the LP has a solution.

We define a cut of the network G a partition $\{W, W'\}$ of V such that $s \in W$ and $t \in W'$. Given a cut $\{W, W'\}$ of G, we define capacity of the cut $\{W, W'\}$ the maximum amount of flow that can pass from W to W':

$$C(W, W') = \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} c_{ij}.$$

Given a feasible flow $\mathbf{f}$ and a cut $\{W, W'\}$ of G, we define the flow of the cut $\{W, W'\}$ the exact amount of flow that passes from W to W':

$$F(W, W') = \overbrace{\sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} f_{ij}}^{\text{flow from } W \text{ to } W'} - \overbrace{\sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W' \\ j\in W}}^{n} f_{ij}}^{\text{flow from } W' \text{ to } W}.$$

Given a feasible flow $\mathbf{f}$ of G, we can show (thanks to the fact that $\mathbf{f}$ is a feasible flow) that for each cut $\{W, W'\}$ of G it holds

$$F(W, W') = v.$$

The flow of the cut $\{W, W'\}$ does then not depend on the cut $\{W, W'\}$, then we can denote it simply by F, dropping the dependence on the specific cut.
Remark that, given a feasible flow $\mathbf{f}$ of G, it holds

$$v \leq C(W, W') \qquad \forall \{W, W'\} \text{ cut of } G. \tag{9.1}$$

Indeed, given a cut $\{W, W'\}$ of G, we have

$$v = F = F(W, W') = \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} f_{ij} - \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W' \\ j\in W}}^{n} f_{ij} \leq \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} f_{ij} \leq \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} c_{ij} = C(W, W').$$

Given a feasible flow $\mathbf{f}$ of G, an edge $(i, j) \in E$ is said saturated if $f_{ij} = c_{ij}$. A backward edge is a couple (j, i) such that $(i, j) \in E$. The edges of G are on the contrary called forward

edges. An backward edge (j,i) is void if $f_{ij}=0$.
A path in the network G is a set

$$P=\{(s,i_1),(i_1,i_2),\dots,(i_p,t)\}$$

whose elements (i_j,i_{j+1}) are direct or backward edges. Given a feasible flow $\mathbf{f}$ in G, a path P in G is an augmenting path if it does not have saturated forward edges and it does not have void backward edges.

Remark 9.2.1. *If $\mathbf{f}$ is a feasible flow in G with value v and P is an augmenting path in G, then it exists an admissible flow $\mathbf{f}_{\text{new}}$ of G with value $v_{\text{new}}>v$.*

Proof. We build $\mathbf{f}_{\text{new}}$ with the following algorithm:

Given $\mathbf{f}$ feasible flow in G (for example $\mathbf{f}=\mathbf{0}$) of value v, P augmenting path in G.

1. Set
$$P_+=\{\text{forward arcs of } P\},\qquad P_-=\{\text{backward arcs of } P\}$$

2. Compute
$$\delta_+=\min\{c_{ij}-f_{ij}\mid i,j=1,\dots,n,\ (i,j)\in P_+\},$$
$$\delta_-=\min\{f_{ij}\mid i,j=1,\dots,n,\ (j,i)\in P_-\}.$$

3. Set $\delta=\min\{\delta_+,\delta_-\}$

4. Set $\forall i,j=1,\dots,n:(i,j)\in E$
$$(\mathbf{f}_{\text{new}})_{ij}=\begin{cases} f_{ij}+\delta & \text{if } (i,j)\in P_+\\ f_{ij}-\delta & \text{if } (j,i)\in P_-\\ f_{ij} & \text{otherwise}\end{cases}$$

Remark that, because P is an augmenting path, it does not have forward saturated edges and void backward edges, then $\delta_+>0$ and $\delta_->0$, so that $\delta>0$.
Remark that $\mathbf{f}_{\text{new}}\in\mathbb{R}^m$ built in this way is a feasible flow (easy to prove TD) and that its value is

$$v_{\text{new}}=\sum_{\substack{j=1\\(s,j)\in E}}^{n}(\mathbf{f}_{\text{new}})_{sj}.$$

Among the forward edges from s, exactly one belongs to P: is the forward edge (s,i_1), then

$$v_{\text{new}}=\sum_{\substack{j=1\\(s,j)\in E\\j\neq i_1}}^{n}(\mathbf{f}_{\text{new}})_{sj}+(\mathbf{f}_{\text{new}})_{s,i_1}=\sum_{\substack{j=1\\(s,j)\in E\\j\neq i_1}}^{n}f_{sj}+f_{s,i_1}+\delta=\sum_{\substack{j=1\\(s,j)\in E}}^{n}f_{sj}+\delta=v+\delta>v.$$

□

Theorem 9.2.1. *Let $\mathbf{f}$ be a feasible flow of G with value v.*

v is a solution of LP $\iff$ *there does not exist an augmenting path in G.*

Proof. ($\Longrightarrow$) By contradiction. Assume there exists an augmenting path in G, then for the previous observation it exists a feasible flow $\mathbf{f}_{\text{new}}$ in G with value $v_{\text{new}} > v$, and so v would not be a solution of LP.

($\Longleftarrow$) Let

$$W = \{s\} \cup \{\text{nodes that can be reached from } s \text{ along an augmenting path}\}, \qquad W' = V \setminus W.$$

By assumption $t \in W'$, then $\{W, W'\}$ is a cut of G.

$$v = F = F(W, W') = \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} f_{ij} - \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W' \\ j\in W}}^{n} f_{ij}.$$

The edges $(i,j) \in E : i \in W, j \in W'$ are such that $f_{ij} = c_{ij}$. By contradiction, let $(i,j) \in E : i \in W, j \in W'$ be such that $f_{ij} < c_{ij}$. Then the edge (i,j) would be non saturated, and, as $i \in W$ is reachable from s along an augmenting path, also j would be reachable from s thorough an augmenting path, so $j \in W$, which is in contradiction with the assumption $j \in W'$.
The edges $(i,j) \in E : i \in W', j \in W$ are such that $f_{ij} = 0$. By contradiction, if it exists $(i,j) \in E : i \in W', j \in W$ such that $f_{ij} > 0$, then the backward edge (j,i) wouldn't be void, and so, because $j \in W$ is reachable from s by an augmenting path, also i would be reachable from s with an augmenting path. Then $i \in W$, in contradiction with $i \in W'$.
Then

$$v = \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W \\ j\in W'}}^{n} c_{ij} - \sum_{\substack{i,j=1 \\ (i,j)\in E \\ i\in W' \\ j\in W}}^{n} 0 = C(W, W').$$

We have then a cut $\{W, W'\}$ of G such that $v = C(W, W')$. From (9.1), the value of the other flows in G cannot be larger than $C(W, W')$, i.e. it must be lower than v, then v is solution to the LP. □

The LP can be solved using the Ford-Fulkerson algorithm:

Given $\mathbf{f}$ feasible flow of G (for example $\mathbf{f} = \mathbf{0}$) of value v

1. Look for an augmenting path G
2. If such a path is not found, then return v and stop.
   Otherwise
   - 2.1 Build $\mathbf{f}_{\text{new}}$ with the algorithm sketched in the previous remark
   - 2.2 Set $v_{\text{new}} = v + \delta$
   - 2.3 Go back to 1

The algorithm returns the value of a flow of G such that there do not exist augmenting paths in G, that is, for the previous theorem, whose value is a solution of the LP.

Because

$$v = \sum_{\substack{j=1 \\ (s,j)\in E}}^{n} f_{sj} \quad \leq \quad \underbrace{\sum_{\substack{j=1 \\ (s,j)\in E}}^{n} c_{sj}}_{\text{is a costant}} := C,$$

and because at each step of the algorithm v is increased of at least 1, the algorithm stops in at most C steps. The cost of a step is mainly related to the cost of searching for an augmenting path in G, i.e. $O(m)$. Then the Ford-Fulkerson algorithm costs $O(Cm)$, it has then a linear complexity (and so polynomial) in m.

Theorem 9.2.2. *Max flow-min cut*
Let $\mathbf{f}$ *be a feasible flow of* G *with value* v*.*

$$v \text{ is a solution to the LP} \iff v = C_{\min},$$

where $C_{\min} = \min\{C(W, W') \mid \{W, W'\} \text{ cut of } G\}$.

Proof. ($\Longrightarrow$) The assumption, from the previous theorem, guarantees that there do not exist augmenting paths in G. Repeating the proof of the implication ($\Longleftarrow$) of the previous theorem, we find a cut $\{\overline{W}, \overline{W}'\}$ of G such that

$$v = C(\overline{W}, \overline{W}'). \tag{9.2}$$

From (9.1) it holds

$$v \leq C(W, W') \qquad \forall \{W, W'\} \text{ cut of } G,$$

then from (1) it holds

$$C(\overline{W}, \overline{W}') \leq C(W, W') \ \ \forall \{W, W'\} \text{ cut of } G,$$

that is $C(\overline{W}, \overline{W}') = C_{\min}$. Then from (9.2)

$$v = C_{\min}.$$

($\Longleftarrow$) Because $C_{\min} = C(W, W')$ for some cut $\{W, W'\}$ of G,

$$v = C(W, W').$$

From (9.1), the value of any other flow of G cannot be larger than $C(W, W')$, that is it cannot be larger than v, then v is solution of the LP.

□

If at step k it holds $\mathbf{p}_k = \mathbf{0}$ and if it exists at least a $j \in \{1, \ldots, M\}$ such that $(\boldsymbol{\lambda}_k)_{i_j} < 0$, said

$$(\boldsymbol{\lambda}_k)_{i_s} = \min\{(\boldsymbol{\lambda}_k)_{i_j} \mid j = 1, \ldots, M : (\boldsymbol{\lambda}_k)_{i_j} < 0\},$$

the algorithm sets

$$\mathbf{x}_{k+1} = \mathbf{x}_k,$$

$$\mathcal{W}(\mathbf{x}_{k+1}) = \mathcal{W}(\mathbf{x}_k) \setminus \{i_s\}$$

and moves on. That is, it finds the solution $\mathbf{p}_{k+1}$ of the problem

$$\begin{aligned}&\min_{\mathbf{p}\in\mathbb{R}^n} \frac{1}{2}\mathbf{p}^T Q\mathbf{p} + \mathbf{g}_{k+1}^T\mathbf{p}.\\ &A_E\mathbf{p} = \mathbf{0}\\ &(A_I\mathbf{p})_i = 0 \ \ \forall i \in \mathcal{W}(\mathbf{x}_{k+1})\end{aligned}$$

It holds

$$\mathbf{p}_{k+1} \neq \mathbf{0}, \qquad (A_I\mathbf{p}_{k+1})_i > 0 \ \ \forall i \in \{1,\dots,m\} \setminus \mathcal{W}(\mathbf{x}_{k+1}).$$

Remark 9.2.2. *We do not prove the theorem, but we show why it is relevant. Once we have found $\mathbf{p}_{k+1} \neq \mathbf{0}$, we set*

$$\mathbf{x}_{k+2} = \mathbf{x}_{k+1} + \alpha\mathbf{p}_{k+1}$$

where we choose $\alpha \in (0,1]$ such that $\mathbf{x}_{k+1} + \alpha\mathbf{p}_{k+1} \in \Omega$, i.e., such that

$$(A_I\mathbf{x}_{k+1})_i + \alpha(A_I\mathbf{p}_{k+1})_i \geq (\mathbf{b}_I)_i \qquad \forall i \in \{1,\dots,m\} \setminus \mathcal{W}(\mathbf{x}_{k+1}). \tag{9.3}$$

Then, because $i_s \in \{1,\dots,m\} \setminus \mathcal{W}(\mathbf{x}_{k+1})$, we choose $\alpha \in (0,1]$ such that

$$(A_I\mathbf{x}_{k+1})_{i_s} + \alpha(A_I\mathbf{p}_{k+1})_{i_s} \geq (\mathbf{b}_I)_{i_s},$$

i.e., as it holds $(A_I\mathbf{x}_{k+1})_{i_s} = (\mathbf{b}_I)_{i_s}$ because $i_s \in \mathcal{W}(\mathbf{x}_k) \subseteq \mathcal{A}(\mathbf{x}_k) \underbrace{=}_{\mathbf{x}_k=\mathbf{x}_{k+1}} \mathcal{A}(\mathbf{x}_{k+1})$, we choose $\alpha \in (0,1]$ such that

$$\alpha(A_I\mathbf{p}_{k+1})_{i_s} \geq 0.$$

The fact that

$$(A_I\mathbf{p}_{k+1})_i > 0 \ \ \forall i \in \{1,\dots,m\} \setminus \mathcal{W}(\mathbf{x}_{k+1})$$

ensures that we will not be forced to choose $\alpha = 0$: this choice, that we eliminate a-priori as we ask $\alpha \in (0,1]$, will imply $\mathbf{x}_{k+2} = \mathbf{x}_{k+1}$, and we would not be able to move from this point.

Lemma 9.2.1. *If at step k it holds $\mathbf{p}_k \neq \mathbf{0}$, then the function*

$$\begin{array}{rccl}\varphi: & (0,1] & \longrightarrow & \mathbb{R}\\ & \alpha & \longmapsto & \varphi(\alpha) = q(\mathbf{x}_k + \alpha\mathbf{p}_k)\end{array}$$

is strictly decreasing.
Then for any $\alpha \in (0,1]$ such that $\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha\mathbf{p}_k$, it will always hold

$$q(\mathbf{x}_{k+1}) < q(\mathbf{x}_k).$$

Proof. $\mathbf{p}_k$ is the only solution of

$$\begin{aligned}&\min_{\mathbf{p}\in\mathbb{R}^n} \widetilde{q}(\mathbf{p}),\\ &A\mathbf{p} = \mathbf{0}\end{aligned} \tag{9.4}$$

where

$$\widetilde{q}(\mathbf{p}) = \frac{1}{2}\mathbf{p}^T Q\mathbf{p} + \mathbf{g}_k^T\mathbf{p}.$$

that is

$$\widetilde{q}(\mathbf{p}_k) < \widetilde{q}(\mathbf{p}) \qquad \forall \mathbf{p} \in \mathbb{R}^n \setminus \{\mathbf{p}_k\} : A\mathbf{p} = \mathbf{0}.$$

Then, because $\mathbf{p}_k \neq \mathbf{0}$ by assumption and it holds $A\mathbf{0} = \mathbf{0}$, it follows

$$\widetilde{q}(\mathbf{p}_k) < \widetilde{q}(\mathbf{0}),$$

i.e.,

$$\frac{1}{2}\mathbf{p}_k^T Q \mathbf{p}_k + \mathbf{g}_k^T \mathbf{p}_k < \mathbf{0},$$

i.e.,

$$\mathbf{g}_k^T \mathbf{p}_k < -\frac{1}{2} \underbrace{\mathbf{p}_k^T Q \mathbf{p}_k}_{\substack{> 0 Q \text{ is pos. def.} \\ \text{and } \mathbf{p}_k \neq \mathbf{0}}} < 0. \tag{9.5}$$

Remark that

$$\begin{aligned} \varphi(\alpha) = q(\mathbf{x}_k + \alpha \mathbf{p}_k) &= \frac{1}{2}(\mathbf{x}_k + \alpha \mathbf{p}_k)^T Q (\mathbf{x}_k + \alpha \mathbf{p}_k) + \mathbf{c}^T (\mathbf{x}_k + \alpha \mathbf{p}_k) \\ = \quad \ldots &= \frac{1}{2}\mathbf{p}_k^T Q \mathbf{p}_k \alpha^2 + \mathbf{p}_k^T \mathbf{g}_k \alpha + q(\mathbf{x}_k) \end{aligned}$$

is a parabola with ?? rivolta verso l'alto (because Q is positive definite) whose vertex has abscissa

$$\alpha_V = \frac{-\mathbf{p}_k^T \mathbf{g}_k}{\mathbf{p}_k^T Q \mathbf{p}_k} \underbrace{>}_{(9.5)} 0.$$

Then $\varphi(\alpha)$ is strictly decreasing in $(0, \alpha_V]$. If we prove that $\alpha_V \geq 1$ we get the thesis.
Because $\mathbf{p}_k$ is a solution of

$$\begin{aligned} &\min_{\mathbf{p} \in \mathbb{R}^n} \frac{1}{2}\mathbf{p}^T Q \mathbf{p} + \mathbf{g}_k^T \mathbf{p}, \\ &A\mathbf{p} = \mathbf{0} \end{aligned}$$

it exists $\boldsymbol{\mu}'_k \in \mathbb{R}^{p+M}$ such that $(\mathbf{p}_k, \boldsymbol{\mu}'_k)$ satisfies the KKT of (9.4), that is such that

$$\begin{cases} Q\mathbf{p}_k + \mathbf{g}_k - A^T \boldsymbol{\mu}'_k = \mathbf{0} \\ A\mathbf{p}_k = \mathbf{0} \end{cases}.$$

Multiplying the first set of equations by $\mathbf{p}_k^T$ we obtain

$$\mathbf{p}_k^T Q \mathbf{p}_k + \mathbf{p}_k^T \mathbf{g}_k - \underbrace{\mathbf{p}_k^T A^T}_{= (A\mathbf{p}_k)^T = \mathbf{0}^T} \boldsymbol{\mu}'_k = \mathbf{0},$$

i.e.,

$$\mathbf{p}_k^T Q \mathbf{p}_k = -\mathbf{p}_k^T \mathbf{g}_k,$$

i.e., $\alpha_V = 1$. □

Lemma 9.2.2. *If at step k it holds $\mathbf{p}_k \neq \mathbf{0}$ and $\alpha = 1$, then $\mathbf{p}_{k+1} = \mathbf{0}$.*

In the following theorem we prove that the active-set method stops in a finite number of steps.

Theorem 9.2.3. *The active-set method stops in a finite number of steps.*

Proof. (a) The algorithm meets $\mathbf{p}_k = \mathbf{0}$ at least every n steps.

Let's prove this Let us assume that $\mathbf{p}_k \neq \mathbf{0}$. If $\alpha = 1$, then from Lemma 2 it holds $\mathbf{p}_{k+1} = \mathbf{0}$. Let us assume that $\alpha \neq 1$. Then, from the definition of the algorithm

$$|\mathcal{W}(\mathbf{x}_{k+1})| = |\mathcal{W}(\mathbf{x}_k)| + 1 = M + 1$$

Analogously

$$\mathbf{p}_{k+1} \neq \mathbf{0}, \alpha \neq 1 \quad \Longrightarrow \quad |\mathcal{W}(\mathbf{x}_{k+2})| = |\mathcal{W}(\mathbf{x}_{k+1})| + 1 = M + 2.$$

$$\vdots$$

$$\begin{aligned}\mathbf{p}_{k+(m-M)-1} \neq \mathbf{0}, \alpha \neq 1 \quad &\Longrightarrow \quad |\mathcal{W}(\mathbf{x}_{k+(m-M)})| = |\mathcal{W}(\mathbf{x}_{k+(m-M)-1})|+1 = M+(m-M) = m \\ &\Longrightarrow \quad \mathcal{W}(\mathbf{x}_{k+(m-M)}) = \{1, \ldots, m\}.\end{aligned}$$

Then $\mathbf{p}_{k+(m-M)}$ will be the solution of the problem

$$\begin{aligned}&\min_{\mathbf{p}\in\mathbb{R}^n} \frac{1}{2}\mathbf{p}^T Q\mathbf{p} + \mathbf{g}_k^T\mathbf{p}, \qquad \text{(PQWp)} \\ &A_E\mathbf{p} = \mathbf{0} \\ &(A_I\mathbf{p})_i = 0 \ \ \forall i = 1, \ldots, m\end{aligned}$$

that is

$$\begin{aligned}&\min_{\mathbf{p}\in\mathbb{R}^n} \frac{1}{2}\mathbf{p}^T Q\mathbf{p} + \mathbf{g}_k^T\mathbf{p}, \qquad \text{(PQWp)} \\ &A\mathbf{p} = \mathbf{0}\end{aligned}$$

where $A \in \mathbb{R}^{(p+m)\times n} = \mathbb{R}^{n\times n}$. The only feasible point is $\mathbf{0}$, and so

$$\mathbf{p}_{k+(m-M)} = \mathbf{0}.$$

Because $m - M \leq m \leq n$ we get the thesis.

(b) When the algorithm meets a point $\mathbf{p}_k = \mathbf{0}$ the working-set will never be equal to $\mathcal{W}(\mathbf{x}_k)$. We can prove this by using the fact that from Lemma 1 it holds $\mathbf{p}_{k+1} \neq \mathbf{0}$ and so from the preposition $q(\mathbf{x}_{k+2}) < q(\mathbf{x}_{k+1}) \underbrace{=}_{\mathbf{x}_{k+1}=\mathbf{x}_k} q(\mathbf{x}_k)$.

From (a) and (b), at least every n steps the algorithm abandons forever a given working-set. As the number of working sets is finite (working-sets are subsets of $\{1, \ldots, m\}$), the algorithm stops after a finite number of steps. □

9.2.1 How to find a starting point of Ω

To find a starting point of Ω we build the following artificial problem:

$$\begin{aligned}&\min_{\mathbf{x}\in\mathbb{R}^n, \mathbf{w}\in\mathbb{R}^p, \mathbf{z}\in\mathbb{R}^m} \sum_{i=1}^{p} w_i + \sum_{i=1}^{m} z_i \\ &(A_E\mathbf{x})_i + \gamma_i w_i = (\mathbf{b}_E)_i \ \ \forall i = 1, \ldots, p \\ &(A_I\mathbf{x})_i + z_i \geq (\mathbf{b}_I)_i \ \ \forall i = 1, \ldots, m \\ &\mathbf{w} \geq \mathbf{0} \\ &\mathbf{z} \geq \mathbf{0}\end{aligned}$$

where $A_E, \mathbf{b}_E, A_I, \mathbf{b}_I$ are the same that appears in the original problem, while, set $\widetilde{\mathbf{x}} \in \mathbb{R}^n$,

$$\gamma_i = \begin{cases} 1 & \text{se } (\mathbf{b}_E)_i - (A_E\widetilde{\mathbf{x}})_i \geq 0 \\ -1 & \text{se } (\mathbf{b}_E)_i - (A_E\widetilde{\mathbf{x}})_i < 0 \end{cases} \quad \forall i = 1, \ldots, p.$$

The feasible set for this problem is

$$\Omega_a = \left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{w} \\ \mathbf{z} \end{pmatrix} \in \mathbb{R}^{n+p+m} \;\middle|\; (A_E\mathbf{x})_i + \gamma_i w_i = (\mathbf{b}_E)_i \;\; \forall i = 1, \ldots, p, \right.$$

$$\left. (A_I\mathbf{x})_i + z_i \geq (\mathbf{b}_I)_i \;\; \forall i = 1, \ldots, m, \quad \mathbf{w} \geq \mathbf{0}, \;\; \mathbf{z} \geq \mathbf{0} \right\}.$$

Let

$$\widetilde{\mathbf{w}} = \begin{pmatrix} \widetilde{w}_1 \\ \vdots \\ \widetilde{w}_p \end{pmatrix}, \qquad \widetilde{w}_i = |(\mathbf{b}_E)_i - (A_E\widetilde{\mathbf{x}})_i| \;\; \forall i = 1, \ldots, p,$$

$$\widetilde{\mathbf{z}} = \begin{pmatrix} \widetilde{z}_1 \\ \vdots \\ \widetilde{z}_m \end{pmatrix}, \qquad \widetilde{z}_i = \max\{(\mathbf{b}_I)_i - (A_I\widetilde{\mathbf{x}})_i, 0\} \;\; \forall i = 1, \ldots, m.$$

It is easy to verify that $\begin{pmatrix} \widetilde{\mathbf{x}} \\ \widetilde{\mathbf{w}} \\ \widetilde{\mathbf{z}} \end{pmatrix} \in \Omega_a$.

Remark that the function $f(\mathbf{w}, \mathbf{z}) = \sum_{i=1}^{p} w_i + \sum_{i=1}^{m} z_i$ on the constraint $\begin{pmatrix} \mathbf{w} \\ \mathbf{z} \end{pmatrix} \geq \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}$ has minimum value in $\begin{pmatrix} \mathbf{w} \\ \mathbf{z} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}$. Then, if $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \Omega_a$, then $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}$ is a solution of the artificial problem. It surely exists $\mathbf{x} \in \mathbb{R}^n$ such that $\begin{pmatrix} \mathbf{x} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \Omega_a$ because this means that it exists $\mathbf{x} \in \mathbb{R}^n$ such that

$$(A_E\mathbf{x})_i = (\mathbf{b}_E)_i \;\; \forall i = 1, \ldots, p, \qquad (A_I\mathbf{x})_i \geq (\mathbf{b}_I)_i \;\; \forall i = 1, \ldots, m,$$

i.e., such that

$$A_E\mathbf{x} = \mathbf{b}_E, \qquad A_I\mathbf{x} = \mathbf{b}_I,$$

i.e., it exists $\mathbf{x} \in \Omega$, that is always true. The set of solutions of the artificial problem is

$$\mathcal{S}_a = \left\{ \begin{pmatrix} \mathbf{x} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix} \in \mathbb{R}^{n+p+m} \mid \mathbf{x} \in \Omega \right\}.$$

The active set method applied to the artificial problem starting from $\begin{pmatrix} \widetilde{\mathbf{x}} \\ \widetilde{\mathbf{w}} \\ \widetilde{\mathbf{z}} \end{pmatrix} \in \Omega_a$ gives as a solution a point $\begin{pmatrix} \mathbf{x}^* \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}$ of $\mathcal{S}_a$. Then we have found a point of Ω, $\mathbf{x}^*$, from which we can start with the active set method on the original problem.