On Some Extensions and Performance of Fast-Lipschitz Optimization

MARTIN JAKOBSSON

Master’s Degree Project
Stockholm, Sweden October 2011

XR-EE-RT 2011:021
Abstract

A huge range of problems in applied sciences such as engineering and economics can be formulated as mathematical optimization problems. In general, these must be solved by iterative methods whose convergence properties to a large extent determines what is achievable. In decentralized applications such as wireless sensor networks, information exchange can be expensive and optimization presents additional challenges. A theory for efficient network and distributed optimization is still in its infancy. Fast-Lipschitz optimization is a recently proposed class of optimization problems, where unique solutions fully determined by a system of equations allow for effective algorithms to solve the problem. This master thesis further investigates F-Lipschitz theory and its possible benefits. In particular, goals were to extend the class, and to compare convergence of the algorithms for the computation of the optimal solution to traditional Lagrangian methods. By carefully proving the main properties of the theory, new results are achieved both through relaxation of existing qualifying conditions and introduction of a new one. New forms, different from the original problem, are also studied. They are shown to belong in the class under mild assumptions. The convergence of a F-Lipschitz method is compared to a gradient method, for problems under the assumption of real eigenvalues. Novel conditions for guaranteed faster performance of the F-Lipschitz method are established. Several possibilities of expanding the theory still remain, and some suggestions are given throughout the thesis.
Acknowledgments

First of all I would like to thank my supervisor, Dr. Carlo Fischione, for all the cheerful support and stimulating guidance throughout this project.

I also want to thank Dr. Ather Gattami and the Talang committee, for awarding me the Talang master thesis scholarship. I have felt privileged to write my thesis at the Automatic Control Laboratory and would like to thank everyone here for your warm welcome and enjoyable company.

Finally, I want to thank my family. Without you, none of this would be possible.
Contents

1 Introduction
2 Notation
3 Norms and Matrices
4 Iterative Methods for Systems of Equations
5 Optimization Theory
6 Distributed Optimization
# Fast-Lipschitz Theory

7.1 Definition .................................................. 37
7.2 Existence and uniqueness of solutions .................. 39
    7.2.1 Proof of main theorem ............................... 40
    7.2.2 Relaxation of max-norm conditions ................. 49
7.3 Fast-Lipschitz and other classes of optimization problems ... 53
7.4 Solving a Fast-Lipschitz problem ....................... 53
7.5 Canonical form .............................................. 55
7.6 Fewer constraints than variables ....................... 58
    7.6.1 Variations of qualifying conditions ............... 59
7.7 Missing variables in cost function ..................... 63
7.8 Examples ................................................... 65

# Convergence Speed Analysis

8.1 Problem formulation ....................................... 71
8.2 First order Lagrangian method ......................... 72
    8.2.1 Conditions for convergence ....................... 73
8.3 Comparison With F-L Method ............................ 77

# Conclusions and Future Work

9.1 Conclusions .................................................. 81
9.2 Future Work ................................................ 81

References ..................................................... 83
1 Introduction

Background and problem definition  The theory of mathematical optimization has a large number of applications in sciences such as engineering, operations research, and economics. It helps decision makers finding the best possible strategy for fulfilling a specified goal. The difficulties lies in translating the real world objective and constraints, into a formal problem that is solvable. Most problems, the ones that are not convex, are in general very hard to solve. Even the convex problems usually don’t have solutions on closed form, wherefore iterative methods are used to find approximates converging to an optimum.

The first applications of optimization were in the planning and design stages of a problem. For example, an optimization problem dimensions a product to withstand some worst-case scenario whereafter the product is built. As the algorithms for (convex) problems have become faster and more robust, it is also possible to continuously solve optimization problems on-line, for example in a running control system.

A relatively new area of interest is Wireless Sensor Networks (WSN). In a WSN, nodes communicate with each other through radio. Each node can perform simple calculations and store data in a memory, they might also have some sensing or actuating equipment. This allows for nodes to be placed where conventional wired networks would be expensive or impossible. The nodes are intended to be cheap, wherefore processing power and accuracy might be low. Instead, the nodes need to cooperate with each other to perform their task. This task will most often be formulated as an optimization problem.

Solving these kind of problems, where central coordination might be limited or non-existent, pose additional difficulties, and the theory for doing so is still immature. As a recent part of this research effort, C. Fischione has proposed a new class of optimization problems [1]. It is called Fast-Lipschitz (F-L), and have unique solutions given by a system of equations. Once a problem is known to be F-L [2], the task of solving it is greatly simplified. This is extra noticeable in WSN applications, and in addition the method and its benefits also applies to centralized optimization.

This master thesis investigates some theoretical properties of the F-L class. There were two separate goals; to investigate possible extensions of the class, and to compare convergence speed of F-L solution methods to a traditional method.
**Thesis outline**  The rest of the thesis is structured as follows:

Chapters 2-5 provide notation and a brief overview of some mathematical concepts used later on.

Chapter 6 describes some of the current methods for distributed optimization, and makes an initial comparison to F-L optimization.

In Chapter 7, the theory of F-L optimization is presented. First, the class is defined and its main properties stated and proved. After that, F-L optimization is related and compared to other types of optimization. Finally, several different forms of the original problem are shown to belong in the class, and examples are given.

Chapter 8 contains an analysis where the convergence speed of F-L optimization is compared to a traditional method.

Finally, Chapter 9 presents final remarks and future work.
2 Notation

Scalars A scalar will be denoted with lower case italic letters, e.g., $c$.

Vectors Vectors are represented by lower case bold letters, such as $\mathbf{a}$ or $\mathbf{x}$. If not otherwise mentioned, vectors are always column vectors and $\mathbf{x} \in \mathbb{R}^{n \times 1}$. The $i^{th}$ component of a vector $\mathbf{x}$ is written $x_i$. The $i^{th}$ component of a vector valued expression will be denoted $[\cdot]_i$, e.g., if $\mathbf{x} = \mathbf{a} + \mathbf{b}$ then $x_i = [\mathbf{a} + \mathbf{b}]_i$.

Special vector notations (for vectors of appropriate sizes) are:

- The zero vector $\mathbf{0}$. The zero vector shares notation with the zero matrix, of which it is a special case.
- The one vector $\mathbf{1}$, where all components are 1.
- The $i^{th}$ unit vector $\mathbf{e}_i$, with all components equal to zero except for the $i^{th}$, which is equal to 1.

Matrices Matrices will be denoted by upper case bold letters, e.g. $\mathbf{A}$. The $(i, j)^{th}$ element of a matrix $\mathbf{A}$ is written $A_{ij}$ or $[\mathbf{A}]_{ij}$ (first index is the row index).

The transpose of a matrix $\mathbf{A}$ is denoted $\mathbf{A}^T$ and fulfills $[\mathbf{A}^T]_{ij} = [\mathbf{A}]_{ji}$.

Special matrix notations (for matrices of appropriate sizes) are

- The zero matrix $\mathbf{0}$.
- The identity matrix $\mathbf{I}$. The components of $\mathbf{I}$ can be written with the Kronecker delta notation,

$$[\mathbf{I}]_{ij} = \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}.$$ 

Summation When summation symbols are used, they usually represent a sum over the components of a vector in $\mathbb{R}^n$. If nothing else is indicated, the sum is assumed to cover all the components, i.e.,

$$\sum_i = \sum_{i=1}^n.$$
Summing over all components except for the $j^{th}$ is denoted
\[ \sum_{i \neq j} \]

**Matrix integration** The integral of a matrix (or vector) should be interpreted as element-wise integration. The result is a matrix (or vector) of the same size as the integrand
\[ \int [M(t)]_{ij} \, dt. \]

**Derivatives** The partial derivatives of a function $f(x) : \mathbb{R}^n \to \mathbb{R}$ can be collected in the $n$-dimensional column vector $\nabla f(x)$. This is the **gradient** of $f(x)$ and
\[ [\nabla f(x)]_i = \nabla_i f(x) = \frac{\partial f(x)}{\partial x_i}. \]

The gradient of a vector valued function $g(x) : \mathbb{R}^n \to \mathbb{R}^m$ is a matrix, whose $j^{th}$ column is the gradient of $g_j(x)$,
\[ [\nabla g(x)]_{ij} = \nabla_i g_j = \frac{\partial g_j(x)}{\partial x_i}. \]

Sometimes it is more natural to use the **Jacobian** of $g(x)$, which is the transpose of the gradient. The Jacobian of $g(x)$ is denoted
\[ Jg(x) = (\nabla g(x))^T. \]
3 Norms and Matrices

This chapter contains a brief summary on some theory and concepts related to matrices. This material is needed in Chapter 4, 7 and 8. For more detailed information on these topics, see e.g., [5].

3.1 Vector norms

A (vector) norm \( \| \cdot \| \) on \( \mathbb{R}^n \) is a mapping that to every vector \( x \in \mathbb{R}^n \) assigns a real number \( \| x \| \) such that:

- \( \| x \| \geq 0 \) for all \( x \in \mathbb{R}^n \) and \( \| x \| = 0 \iff x = 0 \) (positive definiteness).
- \( \| cx \| = |c| \cdot \| x \| \) for all \( c \in \mathbb{R} \) and all \( x \in \mathbb{R}^n \) (positive scalability).
- \( \| x + y \| \leq \| x \| + \| y \| \) for all \( x, y \in \mathbb{R}^n \) (triangle inequality).

One of the most common norms is the Euclidean norm \( \| \cdot \|_2 \) defined as

\[
\| x \|_2 = \left( \sum_{i=1}^{n} x_i^2 \right)^{1/2},
\]

which is the intuitive “distance” between two points in \( \mathbb{R}^2 \) or \( \mathbb{R}^3 \).

A generalization of the Euclidean norm is the \( l_p \)-norm family defined by

\[
\| x \|_p = \sum_{i=1}^{n} |x_i|^p \, 1/p \text{ for } p = 1, 2, 3 \ldots
\]

Two special cases are the \( l_1 \)-norm (other names are one-norm or the Manhattan-norm\(^1\)):

\[
\| x \|_1 = \sum_{i=1}^{n} |x_i|
\]

and the \( l_\infty \)-norm (or “infinity-norm” or “max-norm”) defined by

\[
\| x \|_\infty = \max_{i} |x_i|.
\]

\(^1\)The Manhattan-norm is the distance a car needs to travel between two points in Manhattan, where (almost) every block is rectangular.
Hölder’s inequality  For any two vectors $x$ and $y$ in $\mathbb{R}^n$ we have,

$$|x^T y| = \sum_{k=1}^{n} x_k y_k \leq \|x\|_p \cdot \|y\|_q \quad \text{for} \quad \frac{1}{p} + \frac{1}{q} = 1.$$ 

Hölder’s inequality can be seen as a generalized Cauchy-Schwarz inequality, which is obtained when $p = q = 2$.

### 3.2 Matrix norms

A matrix norm is a mapping that for every matrix $A \in \mathbb{R}^{m \times n}$ assigns a real number $\|A\|$. If $A$ is seen as belonging to $\mathbb{R}^{mn}$ the three properties listed for vector norms applies to $\|A\|$ as well (in fact, the apply to any norm), i.e.

- $\|A\| \geq 0$ for all $A \in \mathbb{R}^{m \times n}$ and $\|A\| = 0 \Leftrightarrow A = 0$ (positive definiteness).
- $\|cA\| = |c| \cdot \|A\|$ for all $c \in \mathbb{R}$ and all $A \in \mathbb{R}^{m \times n}$ (positive scalability).
- $\|A + B\| \leq \|A\| + \|B\|$ for all $A, B \in \mathbb{R}^{m \times n}$ (triangle inequality).

#### Induced norms

In the following we will only use induced norms on square matrices, which are related to vector norms as follows.

Every vector norm $\| \cdot \|$ on $\mathbb{R}^n$ has a corresponding induced matrix norm on $\mathbb{R}^{n \times n}$, also denoted $\| \cdot \|$, defined by

$$\|A\| = \max_{\|x\| = 1} \|Ax\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}.$$ 

The norms induced by the $l_1$- and the $l_\infty$-norm respectively are

$$\|A\|_1 = \max_j \sum_i |A_{ij}|,$$

which is the maximum column sum of absolute values, and

$$\|A\|_\infty = \max_i \sum_j |A_{ij}|,$$

which is the maximum row sum of absolute values.

For all $n \times n$-matrices $A$ and $B$ it holds that

- $\|Ax\| \leq \|A\| \cdot \|x\|$ for all $x \in \mathbb{R}^n$
- $\|AB\| \leq \|A\| \cdot \|B\|$  
- $\|A + B\| \leq \|A\| + \|B\|$
3.3 Spectral radius

First, recall that a square matrix $A$ has an eigenvector $v$ and a corresponding eigenvalue $\lambda$ if

$$Av = \lambda v.$$ 

A $n \times n$ matrix can have at most $n$ distinct eigenvalues. There is a rich theory relating the behavior of a matrix to its eigenvalues, and a concept of special interest in numerical methods is the spectral radius.

The spectral radius of $A$, denoted $\rho(A)$ is the maximum magnitude of its eigenvalues,

$$\rho(A) = \max |\lambda_i| \text{ such that } Ax = \lambda_i x \text{ for some } x \neq 0.$$

The spectral radius indicated how iterated multiplication with the matrix behaves. It can be shown (see e.g. [3, Prop. A21]) that

$$\lim_{k \to \infty} A^k = 0 \iff \rho(A) < 1.$$ 

For a general (square) matrix, the spectral radius is not a matrix norm.

As an example take the matrix

$$C = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$ 

$C \neq 0$ is a upper triangular matrix with zero diagonal entries, wherefore all eigenvalues are zero and $\rho(C) = 0$. This violates the positive definiteness property of norms. However, the spectral radius has close ties to induced matrix norms. For any induced norm, $|| \cdot ||$ is bounded below by the spectral radius,

$$\rho(A) \leq ||A||.$$ 

For any $\epsilon > 0$, there exists an induced norm $|| \cdot ||$ such that $||A|| \leq \rho(A) + \epsilon$ [5, lemma 5.6.10]

3.4 Non-negative matrices and vectors

Matrix inequalities

For any two matrices (or vectors, as a special case) $A$ and $B$ of the same size, $A \geq B$ if $A_{ij} \geq B_{ij}$ for every $i$ and every $j$. The opposite inequality $\leq$, and the strict inequalities $>$ and $<$ are defined analogously.

We say that $A$ is non-negative if $A \geq 0$ and $A$ is positive if $A > 0$. Analogously, $A$ is non-positive if $A \leq 0$ and $A$ is negative if $A < 0$. \(^2\)

\(^2\)For the case when $A$ and $B$ are vectors, this is a special case of generalized inequalities where the ordering cone is $\mathbb{R}_+^n = \{ x : x_i \geq 0 \ \forall i \}$. 

15
that \( A \not\leq B \) does not imply \( A > B \), as in the case of real numbers. For example

\[
\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \not\leq \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}
\quad \text{but} \quad \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \not> \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}.
\]

This situation, where some but not all elements of a set can be compared, is called a partial ordering.

**Properties of non-negative matrices** For any pair of non-negative matrices (including vectors) \( A \) and \( B \), the product \( AB = \sum_k A_{ik} B_{kj} \) is also non-negative (since it is the sum of products of non-negative factors). By the same reasoning,

\[
A \geq 0, B \leq 0 \Rightarrow AB \leq 0
\]

and

\[
A \leq 0, B \leq 0 \Rightarrow AB \geq 0.
\]

If both \( A \) and \( B \) above fulfill their inequalities strictly, so does the corresponding product. For example

\[
A > 0, B < 0 \Rightarrow AB < 0.
\]

A weaker assumption is \( A \geq 0 \) with non-zero rows (at least one element non-zero) and \( B > 0 \) or \( A > 0 \) and \( B \geq 0 \) with non-zero columns. A square matrix \( M \) that is invertible and therefore has full rank will always have non-zero rows/columns.

### 3.5 Matrix series expansion

At several future points we will have a matrix on the form \( M = (I + A) \), where \( A \) is a general square matrix. The following proposition gives useful information on the inverse of \( M \).

**Proposition 1.** Assume \( \rho(A) < 1 \). Then, the inverse of \( M = (I + A) \) can be written as an infinite sum (\( \rho(A) < 1 \) implies \( \lim_{k \to \infty} A^k = 0 \)):

\[
M^{-1} = (I + A)^{-1} = I - A + A^2 - A^3 + \cdots = \sum_{k=0}^{\infty} (-A)^k.
\]

**Proof.** \( M^{-1} \) is the inverse of \( M \) if \( M^{-1} \) exists and \( M^{-1}M = MM^{-1} = I \). First, show that the infinite sum converges to a finite matrix \( M^{-1} \). This can be done by considering some norm \( \| \cdot \| \) of \( M^{-1} \),

\[
\|M^{-1}\| = \| \sum_{k=0}^{\infty} (-A)^k \| \leq \sum_{k=0}^{\infty} \| (-A)^k \| \leq \sum_{k=0}^{\infty} \| (-A) \|^k = \sum_{k=0}^{\infty} \|A\|^k.
\]

16
As discussed in the section on the spectral radius, when $\rho(A) < 1$ we can choose $\|\cdot\|$ such that $\|A\| \leq \rho(A) + \epsilon < 1$. Therefore, through the geometric series, we can write

$$\|M^{-1}\| = \sum_{k=0}^{\infty} \|A\|^k = \frac{1}{1 - \|A\|}.$$ 

Since the denominator is (strictly) positive, $\|M^{-1}\| < \infty$ and $M^{-1}$ is bounded.

It remains to show that $M^{-1}$ multiplied by $M$ (form right and left) is the identity matrix. Multiplying from the left gives

\[
MM^{-1} = (I + A) \sum_{k=0}^{\infty} (-A)^k = I + \sum_{k=1}^{\infty} (-A)^k + (-1) \sum_{k=0}^{\infty} (-A)^{k+1} = I + \sum_{k=1}^{\infty} (-A)^k - \sum_{k=1}^{\infty} (-A)^k = I.
\]

As $A$ and $\sum_{k=0}^{\infty} (-A)^k$ commute, multiplying from the right gives the same result.

**Corollary.** Assume $A \geq 0$. Then, $(I - A)^{-1} \geq 0$.

**Proof.** Proposition 1 gives

\[
(I - A)^{-1} = (I + (-A))^{-1} = \sum_{k=0}^{\infty} (-(-A))^k = \sum_{k=0}^{\infty} A^k.
\]

Every term of the infinite sum is a product of non-negative matrices wherefore the terms are non-negative. Therefore, being the sum of only non-negative terms, $(I - A)^{-1}$ is non-negative.
4 Iterative Methods for Systems of Equations

System of equations without an analytical solution can still be solved with iterative methods. A sequence of approximate solutions \( \{x^k, k = 0, 1, 2, \ldots \} \), where \( x^{k+1} \) is a function of previous iterates (usually just \( x^k \), but higher order methods can use any number of previous iterates). There are two main strategies to show convergence, based on decent methods and contraction mappings respectively.

In the decent method case, one shows that some function of the iterates form a decreasing sequence bounded from below. The function is designed to attain this lower bound only at the sought solution, wherefore the sequence \( \{x^k\} \) must converge to the intended point. This is closely related to descent methods in optimization and to Lyapunov stability in control.

The other method, uses contraction mappings to show that the distance from an iterate and the solution decreases in each iteration. A brief definition to contraction mappings is given below. An extensive treatment can be found in [3, 8].

**Contraction Mappings** A contraction mapping \( T(\cdot) \) is a function from a subset of \( \mathbb{R}^n \) to itself, such that \( T : D \to D \) fulfills
\[
||T(x) - T(y)|| \leq \alpha ||x - y|| \quad \forall x, y \in D
\]
for some norm || · ||. Here \( \alpha \) is called the modulus of the contraction and fulfills \( \alpha < 1 \). Contraction mappings are always continuous, this follows directly from the definition.

Any vector \( x^* \) such that \( T(x^*) = x^* \) is called a fixed point of \( T \).

The following proposition can be seen as a special case of the Banach Fixed Point theorem.\(^3\)

**Proposition 2.** [3, sec 3.1.1] If \( D \subseteq \mathbb{R}^n \) is closed, and \( T : D \to D \) is a contraction, then \( T \) has a unique fixed point \( x^* \). The sequence \( x^{k+1} = T(x^k) \) converges to \( x^* \) geometrically:
\[
||x^k - x^*|| \leq \alpha^k ||x^0 - x^*|| \quad \forall k \geq 0.
\]

---

\(^3\)The Banach Fixed Point theorem considers any metric space \((X, d)\). Here we assume \( X \subseteq \mathbb{R}^n \) and \( d(x, y) = ||x - y|| \) for some norm || · ||.
5 Optimization Theory

The purpose of this section is to provide some background and notation in optimization. As this is a huge branch of applied mathematics the treatment here will neither be detailed, nor cover anything but the most basic subjects. Standard references are [7] for convex optimization and [8] for non-linear optimization in general.

5.1 General optimization

The general optimization problem consists of choosing an element from a set minimizing some function.

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad x \in X
\end{align*}
\]

Here, \( f(x) \) is called the cost function (or objective function) and \( X \) is called the feasible set. The solution of the problem is an element \( x^* \) belonging to \( X \), such that all other elements \( y \) in \( X \) give a higher cost (the optimal cost is usually denoted \( p^* \)) when compared to \( x^* \),

\[
p^* = f(x^*) \leq f(y) \quad \forall y \in X.
\]

Any \( x^* \) for which this holds is called an optimal point. If the inequality above holds strictly, and any other point than \( x^* \) gives a higher cost, \( x^* \) is said to be the unique optimum, or uniquely optimal. Optimal in this sense can be clarified as globally optimal, as opposed to locally optimal. A point \( \bar{x} \) is locally optimal if it is a minimizer of \( f \) (restricted to the feasible set) in a neighborhood of \( \bar{x} \). This means, that for some \( \epsilon > 0 \) and some metric \( d \), \( \bar{x} \) is an optimal point of the the problem

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad x \in X \\
& \quad d(\bar{x}, x) \leq \epsilon.
\end{align*}
\]

The feasible set is typically expressed through a set of equality and inequality constrained functions of \( x \). Such problems can be written in a standard form, called canonical form, as

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g(x) \leq 0 \\
& \quad h(x) = 0.
\end{align*}
\]
The \( i^{th} \) component of \( g \), called the \( i^{th} \) inequality constraint, is said to be \textit{active} at a point \( x \) if the inequality holds strictly, and \( g_i(x) = 0. \)\(^4\)

Although the standard optimization problem is to minimize a cost function, it might instead be more natural to maximize an objective. This needs no additional theory, since two problems

\[
p_1^* = \max_x f(x) \quad \text{and} \quad p_2^* = \min_x -f(x)
\]

are closely related; They have the same set of optimal points and \( p_1^* = -p_2^* \). The F-Lipschitz class of problems discussed in this report involves the maximization of an objective function. Even though it might be more natural to denote the optimal objective function value as \textit{gain} or \textit{yield}, we will stick \textit{cost}. That is, our goal will be to find the \( x \) giving the highest cost.

5.2 Convex optimization

5.2.1 Convex analysis

**Convex sets** A set \( X \subseteq \mathbb{R}^n \) is convex if any pair of its elements can be connected with a straight line entirely in \( X \). That is, \( X \) is convex if and only if the point

\[
z = (1 - \alpha)x + \alpha y
\]

lies in \( X \), for any \( x, y \in X \) and \( 0 \leq \alpha \leq 1 \).

If \( X \) is expressed on canonical form,

\[
X = \{ x : g(x) \leq 0, h(x) = 0 \}
\]

then \( X \) is convex if \( g(x) \) is convex and \( h(x) \) is affine.

**Convex functions** A scalar valued function defined on a convex set \( X \)

\[
f(x) : X \subseteq \mathbb{R}^n \to \mathbb{R}
\]

is convex if, for any \( x, y \in X \) and \( 0 < \alpha < 1 \),

\[
(1 - \alpha) f(x) + \alpha f(y) \geq f((1 - \alpha) x + \alpha y).
\]

The function \( f(\cdot) \) is guaranteed to be defined for \( z = (1 - \alpha)x + \alpha y \) since \( X \) is convex. If inequality (1) holds strictly, \( f(\cdot) \) is said to be \textit{strictly convex}. A convex function is always lower bounded by its tangent plane and upper bounded by any secant.

\(^4\)When a problem is convex, constraints that are not active at a optimal point can be removed without changing the solution to the problem.
Switching sign of the inequality of equation (1) gives the definition of a concave function. This concept is almost analogous to convexity. For example, if \( f \) is convex on the convex set \( X \), then \( -f \) is concave on \( X \).

Convex analysis provides us with a rich theory describing when functions are convex. For example:

- A function \( f(x) \) on \( X \) is convex if and only if \( f \) restricted to any line within \( X \) is convex.

- For a function \( f(x) \) of a scalar variable on an open domain \( X \), \( f \) is convex if and only if \( f''(x) \geq 0 \) for every \( x \in X \). If the inequality is strict, \( f \) is strictly convex.

- A non-negative combination of convex functions is convex.

- The supremum of a set of convex functions is convex.

- The infimum of a set of concave functions is concave.

This is of course only a small fraction of what can be said, for an in-depth treatment see [7, 6]

### 5.2.2 Convex optimization

The convex optimization problem consists of minimizing a convex cost function over a convex set. The standard form of writing this is

\[
\begin{align*}
\text{min} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0 \quad i = 1, \ldots, m \\
& \quad h_i(x) = 0 \quad i = 1, \ldots, p
\end{align*}
\]

Here, \( f, g, h : \mathbb{R}^n \rightarrow \mathbb{R} \) are scalar functions where \( f \) and \( g_i \) are convex while \( h_i \) are affine. Maximizing a concave function on a convex set can easily be formulated as a convex problem, as \( \max f = -\min(-f) \) and \( -f \) is convex if \( f \) is concave. Because of this, the maximization of a concave function is usually referred to as a convex optimization problem.

The most important property of a convex optimization problem is that every local minimizer is also a global minimizer. This greatly simplifies the problem, since one only need local information to determine whether a point is locally (and therefore globally) optimal or not.

For the unconstrained convex problem \( \min_x f(x) \), \( x^* \) is an optimal point if and only if \( \nabla f(x^*) = 0 \). This the basic condition of a stationary point. For constrained problems, optimality conditions usually involve extra variables (multipliers), see the section on duality.
5.3 Solving a convex problem

Solving a convex problem is usually easier than solving non-convex problems. In rare cases, optimality conditions such as vanishing gradient for unconstrained problem, or the KKT conditions (see section on duality) give a closed form solution. When this is not the case, several well studied iterative algorithms gives approximate solutions within specified tolerances.

The rest of this section contains a (very) brief overview of some different types of methods. Subgradient methods receive a little more attention since they are an important part of many distributed strategies.

5.3.1 Descent methods

As any local optimum also is globally optimal, one strategy is to generate a sequence of points with decreasing cost function value. These algorithms are called descent methods. As long as the problem is solvable (and the cost function is bounded from below), any such sequence will converge to an optimal point. Directions of decreasing cost (called descent directions) can be found the gradient of the cost function. These are called gradient methods and special variants are Steepest descent (where direction is given by the negative gradient), and the more sophisticated Newton method. Newton’s method has the drawback of requiring second derivatives (the Hessian for functions of a vector variable), which must also be inverted. As this is computationally demanding, Newton-like methods that replaces the Hessian with a more easily invertible approximate of Hessian have been developed.

When the function is not differentiable, gradients can often be replaced by subgradients (see, e.g., [8]).

5.3.2 Interior point algorithms

Interior point methods solve constrained convex problems by considering a sequence of unconstrained problems, where infeasibility translates to a penalty in the objective function. These (sub-) problems are typically solved with Newton-like methods, and their optimal point is used as an initial point for the next iteration. For every step an approximation of the optimal point is achieved, and the as approximation becomes better, the solutions will converge to the optimal point of the original problem. For every iterate along the way, sub-optimality can be bounded with duality techniques.

Interior point methods are not as straightforward to implement, but their fast convergence makes them the method of choice for centralized optimization, where ready-made solvers exist in, e.g., Matlab.

More information on interior point methods can be found in [7].
5.3.3 Subgradient method

This subsection contains subgradients and subgradient methods, used to generalize gradient methods to non-differentiable functions. For a more detailed description, see [8, 12].

Subgradient definition  Given a convex function $f(x) : \mathbb{R}^n \to \mathbb{R}$ defined on the convex set $X \subseteq \mathbb{R}^n$, a subgradient (or subgradient) at $x$ is any vector $v$ such that for all $y \in X$,

$$f(y) - f(x) \geq v^T(y - x).$$

The set of all subgradients of $x$ are called the subdifferential of $x$ and denoted $\partial f(x)$. When $f$ is differentiable, the gradient of $f$ is a (the unique) subgradient, $\nabla f \in \partial f$. The sub-differential is always non-empty and convex, i.e., if $v$ and $y$ are subgradients of $f$ at $x$, so is $z = (1 - \alpha)v + \alpha y$ for $0 \leq \alpha \leq 1$.

Written as

$$f(y) \geq g(y) = f(x) - v^T x + v^T y,$$

it is apparent that a subgradient at $x$ describes an affine global underestimator of $f(y)$ (with equality only when $y = x$).

When $v = 0 \in \partial f(\bar{x})$, $g(y) = f(\bar{x})$ and $f(y) \geq f(\bar{x})$ for all $y \in X$, i.e. $\bar{x}$ is the global minimizer of $f$. The converse of this is also true, although not as easily demonstrated.

The subgradient algorithm  The general algorithm for minimizing a convex function $f(x)$ generates a sequence $\{x^k\}$, calculated as

$$x^{k+1} = x^k - \alpha_k g^k,$$

where $g^k \in \partial f(x^k)$ is any subgradient of $f$ at $x^k$ and $\alpha_k > 0$ is the $k^{th}$ step size suitably chosen by some step size rule.

There are many step size rules that under additional (quite natural, such as finite subgradients) assumptions on the problem guarantee convergence at least to a set of $\epsilon$-suboptimal set of points. One such rule, that guarantees convergence to the optimal point, is to chose $\alpha_k > 0$ such that

$$\sum_{k=0}^{\infty} \alpha_k^2 < \infty \quad \text{while} \quad \sum_{k=0}^{\infty} \alpha_k = \infty.$$

Convergence is in general slow and good stopping criteria are usually hard to find, but the merits of this method lies in the ability to handle non-differentiable problems.
5.4 Duality

Duality is a powerful concept in optimization theory that, by introducing extra variables, allows for the study of a related, often simpler problem. Duality theory makes no assumptions on convexity of the original problem, although this often gives stronger results. Throughout this section we will consider a problem on canonical form,

\[
\begin{align*}
\min & \quad f_0(x) \\
\text{s.t.} & \quad f_i(x) \leq 0 \quad i = 1, \ldots, m \\
& \quad h_i(x) = 0 \quad i = 1, \ldots, p
\end{align*}
\]

(2)

5.4.1 The Lagrangian and the dual function

Introduce the Lagrangian as

\[
L(x, \lambda, \nu) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{p} \nu_i h_i(x).
\]

An alternative formulation, with \( f(x) = [f_1(x), \ldots, f_m(x)]^T \) and \( h(x) = [h_1(x), \ldots, h_p(x)]^T \), is

\[
L(x, \lambda, \nu) = f_0(x) + \lambda^T f(x) + \nu^T h(x).
\]

The new variables \( \lambda \geq 0 \in \mathbb{R}^m \) and \( \nu \in \mathbb{R}^p \) are called Lagrange multipliers or dual variables.

The dual function \( g(\lambda, \nu) \) of the problem is obtained by minimizing the Lagrangian over all \( x \) (where the functions \( f_0, f_i \) and \( h_i \) are defined),

\[
g(\lambda, \nu) = \inf_{x} L(x, \lambda, \nu) = \inf_{x} \left( f_0(x) + \lambda^T f(x) + \nu^T h(x) \right).
\]

The function \( g(\lambda, \nu) \) is the infimum of a family of affine functions (parametrized by \( x \)). Therefore, regardless of whether the original problem is convex or not, the dual function is concave.

Denote \( p^* \) as the optimal value of (2). Since \( x^* \), the (possibly non-unique) minimizer of \( f_0 \), must be feasible we have \( f(x^*) \geq 0 \) and \( h(x^*) = 0 \). Therefore (for any \( (\lambda, \nu) \) where \( \lambda \geq 0 \)),

\[
L(x^*, \lambda, \nu) = f_0(x^*) + \lambda^T f(x^*) + \nu^T h(x^*) \leq f_0(x^*) = p^*
\]

and

\[
g(\lambda, \nu) = \inf_{x} L(x, \lambda, \nu) \leq L(x^*, \lambda, \nu) \leq f_0(x^*) = p^*.
\]

In other words, for any pair of feasible dual variables \( \lambda \geq 0 \), \( g(\lambda, \nu) \) is a lower bound on \( p^* \), the optimal value of the original problem.
5.4.2 The dual problem

The goal of the dual problem is to find the highest lower bound of \( p^* \), i.e.

\[
d^* = \max_{\lambda \geq 0, \nu} g(\lambda, \nu).
\]

Any pair \((\lambda^*, \nu^*)\) (not necessarily unique) that optimized is called dual optimal (or optimal Lagrange multipliers). Since \( g \) is concave regardless of the original problem, the dual problem is always a convex optimization problem.

5.4.3 Weak duality

Even though \( d^* \) represents the highest lower bound for \( p^* \), it still holds that

\[
d^* \leq p^*.
\]

This is known as weak duality, and is of great use since it bounds the suboptimality of any given point in original problem. For example, having found the optimal value of the dual problem to be \( d^* \), and given a primal feasible \( \tilde{x} \) one knows that

\[
d^* \leq p^* \leq f_0(\tilde{x}).
\]  \[(3)\]

If \( d^* \) and \( f_0(\tilde{x}) \) are close enough, then so are \( p^* \) and \( f_0(\tilde{x}) \) and \( \tilde{x} \) can be taken as a good enough solution of the primal problem. In fact, if \( d^* = f_0(\tilde{x}) \), then \( (3) \) must hold with equality and \( \tilde{x} \) is optimal. In practice, since is very uncommon with a closed form solution, this provides an important stopping criterion for many iterative techniques.

5.4.4 Strong duality and the KKT conditions

The difference \( p^* - d^* \) is known as the duality gap of the problem. When \( p^* = d^* \) and the duality gap is zero, strong duality is said to hold. Strong duality has far reaching implications. For any pair of (primal and dual) optimal points \( x^* \) and \( (\lambda^*, \nu^*) \), strong duality and the definition of the dual function gives

\[
f_0(x^*) = g(\lambda^*, \nu^*)
\]

\[
= \inf_{x} \left( f_0(x) + \lambda^*^T f(x) + \nu^*^T h(x) \right)
\]

\[
\leq f_0(x^*) + \lambda^*^T f(x^*) + \nu^*^T h(x^*) \leq f_0(x^*),
\]  \[(4)\]

Since \( f_0(x^*) = \cdots \leq f_0(x^*) \), all the inequalities above must hold with equality. This means that

\[
\inf_{x} L(x, \lambda^*, \nu^*) = \inf_{x} \left( f_0(x) + \lambda^*^T f(x) + \nu^*^T h(x) \right)
\]

\[
= f_0(x^*) + \lambda^*^T f(x^*) + \nu^*^T h(x^*),
\]
i.e., \( x^\star \) is a (not necessarily the unique) minimizer of the Lagrangian \( L(x, \lambda^\star, \nu^\star) \).

If \( f, h, \) and \( f_0 \) are differentiable, then the Lagrangian must have a stationary point at \( x^\star \), i.e.
\[
\nabla_x L(x^\star, \lambda^\star, \nu^\star) = 0.
\]

The last inequality of (4) holds with equality only if
\[
\lambda^\star^T f(x^\star) + \nu^\star^T h(x^\star) = 0.
\]

If \( x^\star \) is optimal it must also be feasible, wherefore \( h(x^\star) = 0 \) and \( f(x^\star) \leq 0 \).

The second term is therefore always zero, and the first term can be written
\[
\sum_{i=1}^{m} \lambda_i f_i(x^\star).
\]

For every \( i \), \( \lambda \geq 0 \) and \( f(x^\star) \leq 0 \) gives \( \lambda_i f_i(x^\star) \leq 0 \). A sum with only non-positive terms is zero only if all terms are zero, i.e.

\[
\lambda_i f_i(x^\star) = 0 \quad \forall i.
\]

This is known as complementary slackness. If an inequality constraint is not active, the dual variable associated with this constraint must be zero.

For any optimization problem where strong duality holds, the observations above together with feasibility requirements for the primal and dual variables give necessary conditions for optimality. These are known as the KKT (Karush-Kuhn-Tucker) conditions:

**Proposition** (KKT Necessary Conditions). *If strong duality holds for problem (2) and \( f_0, f \) and \( h \) are differentiable, then the pair \( \bar{x} \) and \( (\bar{\lambda}, \bar{\nu}) \) are primal and dual optimal only if*

\[
\begin{align*}
\nabla f_0(\bar{x}) + \nabla f(\bar{x})\bar{\lambda} + \nabla h(\bar{x})\bar{\nu} &= 0 \\
\lambda_i f_i(x^\star) &= 0 \quad i = 1, \ldots, m.
\end{align*}
\]

*Remark.* The first two conditions are primal feasibility and the third condition is dual feasibility. The forth condition is the expanded form of \( \nabla_x L(x, \lambda, \nu) = 0 \) and the fifth is complementary slackness.

The KKT conditions only apply when strong duality holds. In general, they only give necessary conditions.

For convex problems however, the KKT conditions are also sufficient. In other words, if a problem is convex and strong duality holds, then a pair of points (primal and dual) are optimal if and only if they satisfy the KKT conditions.
Strong duality often holds for convex problems. When a problem is convex it is often also strongly dual. Conditions that, in addition to convexity, guarantee strong duality are called constraint qualifications. One of the simplest and most common constraint qualifications is Slater’s Condition.

Proposition (Slater’s Condition). \([7, \S 5.2.3]\) If problem \((\mathcal{P})\) is convex, and there exists at least one point \(\tilde{x}\) such that  
\[
\begin{align*}
    h(\tilde{x}) &= 0 \\
    f(\tilde{x}) &< 0,
\end{align*}
\]
then strong duality holds and the KKT conditions are both necessary and sufficient for optimal points.

5.5 Vector optimization

Consider again the general problem  
\[
\begin{align*}
    \min & \quad f_0(x) \\
    \text{s.t.} & \quad x \in X
\end{align*}
\]
when the cost function \(f_0 : \mathbb{R}^n \rightarrow \mathbb{R}^m\) is multivalued. This is the case when one wants to minimize several different objectives simultaneously. Comparing the cost of different points is no longer straightforward. Let \(\mathcal{O} \subseteq \mathbb{R}^m\) be the set of objective values of all points in the feasible set:  
\[
\mathcal{O} = \{f_0(x) : x \in X\}.
\]
\(\mathcal{O}\) can be partially ordered by the positive orthant \(\mathbb{R}^m_+ = \{u : u_i \geq 0 \ \forall i\}\). We write \(u \succeq v\) if \(u - v \in \mathbb{R}^m_+\), i.e., \([u - v]_i \geq 0\) for all \(i\). When wanting to minimize the cost it is possible that \(u \not\leq v\) without \(u > v\).

A feasible point \(x \in X\) is said to be Pareto optimal if there is no other feasible point \(y \in X\) such that \(f_0(y) \leq f_0(x)\). This means that a Pareto optimal point is a point from where it is impossible to lower the cost any component, without raising the cost of some other component.

Pareto optimal points can be found by solving the scalarized problem  
\[
\begin{align*}
    \min & \quad \mu^T f_0(x) \\
    \text{s.t.} & \quad x \in X
\end{align*}
\]
Here, \(\mu > 0\) assigns weights to the components of \(f_0\) and combines them to a single cost, comparable between feasible points. For more information on partial ordering and Pareto optimality, see \([6]\) and \([7]\) respectively.

\(^5\)Note that the inequality constraints should hold with strict inequality. Such a point \(\tilde{x}\) is often called strictly feasible.
6 Distributed Optimization

Research efforts in distributed optimization date back at least to the early 60’s, with decomposition methods for linear programs (e.g. Dantzig-Wolfe decomposition). The goal of these efforts were to tackle problems that previously had been too large to solve efficiently. The idea is to divide a large problem into several smaller ones, and benefits come both from distributing computations over several processors and the fact that problem complexity often grows super-linearly (solving $n$ problems of size $M/n$ is takes less effort than solving one problem of size $M$).

Today, in many applications the reason for distributing optimization have instead shifted towards distribution in the sense of decentralizing the problem. This is required in situations where central coordination is expensive or even impossible. Examples include wireless sensor networks, as described in the introduction, and robotic systems with autonomous agents.

In the rest of this section, three distributed methods are briefly covered. Then, a short discussion of these methods (and their applications to WSNs), will serve as a short introduction (and motivation) of F-L optimization. An extensive overview of distributed methods in general can be found in [3], while [9, 10, 11, 12] contain selected topics.

6.1 Decomposition methods

The trivial case is a problem on the form

$$p^* = \min_{x=(x_1, x_2)} f(x) = f_1(x_1) + f_2(x_2)$$

subject to $x_1 \in X_1$, $x_2 \in X_2$.

This problem is called separable since the two problem parts have no effect on each other. It is possible to solve each part separately and then combining the solutions, $p^* = p_1^* + p_2^*$ and $x^* = (x_1^*, x_2^*)$ where

$$p_1^* = \min_{x_1} f_1(x_1) \quad \text{and} \quad p_2^* = \min_{x_2} f_2(x_2)$$

subject to $x_1 \in X_1$ \quad \text{subject to} \quad x_2 \in X_2$.

Decomposition is also possible (but no longer trivial) when the sub problems share a common variable or when they are restricted by a common
In the two following sub-sections, primal and dual decomposition is applied to the simple example of a common variable. This is only intended to compare the basic features of the two methods.

6.1.1 Primal decomposition

Consider first the unconstrained problem with a shared variable,

$$\min_{x=(x_1,x_2,y)} f_1(x_1,y) + f_2(x_2,y)$$

or

$$\min_{x=(x_1,x_2)} f_1(x_1) + f_2(x_2)$$

subject to \( h_1(x_1) + h_2(x_2) \leq 0 \).

If \( f_1 \) and \( f_2 \) are convex, so is \( \phi_1(y) + \phi_2(y) \). Problem (5) is called the master problem, and is typically solved iteratively using subgradient methods through the following algorithm:

**Algorithm**  Repeat:

*Solve sub-problems:* For a given \( y \) and for each \( i \), find a \( x_i \) that minimizes \( f_i(x_i,y) \) and a subgradient \( v_i \) of \( \phi_i \) at \( y \). If possible, the subproblems are solved simultaneously.

*Update complicating variable:* If \( v_1 \) and \( v_2 \) are subgradients of \( \phi_1 \) and \( \phi_2 \), then \( v_1 + v_1 \) is a subgradient of \( \phi_1 + \phi_2 \). The subgradient method now gives

$$y^{k+1} = y^k - \alpha_k(v_1 + v_2).$$
Every iteration generates a $x^k = (x_1^k, x_2^k)$ that is feasible in the original problem. The cost function $f(x^k)$ does not necessarily decrease at each step, but (with an appropriate step size rule) $x^k \to x^*$ as $k \to \infty$.

6.1.2 Dual decomposition

Once again, consider the problem

$$\min_x f_1(x_1, y) + f_2(x_2, y)$$

s.t. $x_1 \in X_1$, $x_2 \in X_2$.

Assume the problem is strictly convex and that strong duality holds. For simplicity, let the constraints be implicitly given by the cost function, e.g. by redefining

$$f_i(x_i, y) := \begin{cases} f_i(x_i, y) & \text{if } x_i \in X_i \\ \infty & \text{otherwise} \end{cases}$$

The trick of dual decomposition is to introduce local versions $y_1$ and $y_2$ of the complicating variable. If these local variables are required to be equal, the new problem

$$\min_x f_1(x_1, y_1) + f_2(x_2, y_2)$$

s.t. $y_1 = y_2$.

is equivalent. The Lagrangian of this new problem is

$$L(x, \lambda) = f_1(x_1, y_1) + f_2(x_2, y_2) + \lambda^T(y_1 - y_2).$$

The dual function is $g(\lambda) = g_1(\lambda) + g_2(\lambda)$ where

$$g_1(\lambda) = \min_{x_1, y_1} \left( f_1(x_1, y_1) + \lambda^T y_1 \right)$$

and

$$g_2(\lambda) = \min_{x_2, y_2} \left( f_2(x_2, y_2) - \lambda^T y_2 \right).$$

In dual decomposition, the master problem is to maximize the dual function,

$$\max_{\lambda \neq 0} g_1(\lambda) + g_2(\lambda).$$

Again, this is done by the subgradient method. In dual decomposition, finding a subgradient for the master problem is easy. If $(\tilde{x}_1, \tilde{y}_1)$ and $(\tilde{x}_2, \tilde{y}_2)$ are minimizers of the first and second sub-problems respectively, it can be shown that $(\tilde{y}_2 - \tilde{y}_1)$ is a subgradient of the dual function $g$ at $\lambda$. Therefore the algorithm for dual decomposition with subgradient methods is

33
Algorithm  Repeat:

Solve sub-problems: For the given $\lambda$, find a $(\tilde{x}_1, \tilde{y}_1)$ that minimizes (6) and a $(\tilde{x}_2, \tilde{y}_2)$ that minimizes (7). If possible, the sub-problems are solved simultaneously.

Update complicating variable: A subgradient of $g$ at $\lambda$ is $(\tilde{y}_2 - \tilde{y}_1)$, wherefore the subgradient method for maximizing the mast problem gives the next $\lambda$ as

$$\lambda^{k+1} = \lambda^k - \alpha_k (\tilde{y}_2 - \tilde{y}_1).$$

The method solves the dual problem, by adjusting $\lambda$ until $(\tilde{y}_2 - \tilde{y}_1)$ reaches zero (if the master problem has the zero vector as a subgradient, then it is optimal). However, this happens only asymptotically, and until it does $\tilde{y}_2 - \tilde{y}_1 \neq 0$ so the primal problem is not feasible.

At iteration $k$, it is possible to construct a feasible point by averaging $y_1^k$ and $y_2^k$ and solving the related sub-problems of primal decomposition

$$\bar{x}_k^i = \arg \min_{x_i} f_i(x_i, \bar{y}^k) \quad \text{where} \quad \bar{y}^k = \frac{y_1^k + y_2^k}{2}.$$

This point will give an upper bound on the optimal value of the original problem,

$$p^* \leq f(\bar{x}^k) = f_1(\bar{x}_1^k, \bar{y}^k) + f_2(\bar{x}_2^k, \bar{y}^k).$$

The dual function $g(\lambda^k)$ always gives a lower bound on $p^*$, wherefore

$$g(\lambda^k) \leq p^* \leq f(\bar{x}^k).$$

6.2 Alternating Direction Method of Multipliers

Alternating Direction Method of Multipliers (ADMM) is an algorithm that solves problems on the form

$$\min_{x, z} f(x) + g(z) \quad \text{s.t.} \quad Ax + Bz = c,$$

where $f$ and $g$ are convex and strong duality is assumed to hold. This is a (linearly) equality-constrained problem where the variable vector has been partitioned, such that the cost function terms only depend on one of the two halves.

Just like dual decomposition, ADMM solves the dual problem through subgradient methods. However, in order to improve robustness and convergence, ADMM forms the dual function by minimizing the Augmented Lagrangian

$$L_\rho(x, z, \lambda) = f(x) + g(z) + \lambda^T(Ax + Bz - c) + \frac{\rho}{2}||Ax + Bz - c||_2^2.$$
This can be seen at the Lagrangian for the related problem

$$
\min_{x,z} \ f(x) + g(z) + \frac{\rho}{2} ||Ax + Bz - c||^2
$$

s.t. \ Ax + Bz = c.

When this problem is feasible, the added term is zero wherefore the problems are equivalent. The dual function is

$$
g(\lambda) = \min_{x,z} L_\rho(x, z, \lambda).
$$

As in dual decomposition, a subgradient of $g$ at $\lambda$ is given by $(Ax^* + Bz^* - c)$ where $x^*$ and $z^*$ that minimizes the Lagrangian. Different is ADMM however, is that the minimization is divided into two steps (a Gauss-Seidel variant). The algorithm becomes:

**Algorithm**  
Repeat:

$$
x^{k+1} = \arg \min_x L_\rho(x, z^k, \lambda^k)
$$

$$
z^{k+1} = \arg \min_z L_\rho(x^{k+1}, z, \lambda^k)
$$

$$
\lambda^{k+1} = \lambda^k + \rho \left( Ax^{k+1} + Bz^{k+1} - c \right)
$$

When it is possible to formulate problems such that any of the steps above are separable, this step should be performed in a parallel manner.

When $f$ and $g$ are closed convex functions, and strong duality holds for (8), the method will converge (see [10]). As ADMM primarily is intended as a robust and implementable method, convergence of typical problems is quite slow. If only modest accuracy is required (as is the case of many applications), this need not be a problem.

### 6.3 Why Fast-Lipschitz optimization

The main restriction for the methods above are that they can only solve convex problems. Dual decomposition and ADMM also require strong duality and even then, recovery of the primal solution from the dual iterate can be challenging.

In a WSN, the subproblems can be distributed over the nodes. This is a good start, but the master problem still remains. In rare cases this to might be separable and distributed among the nodes, but most often it has to be solved in a centralized manner. Even if the master problem is easy enough to be implemented in one of the nodes (which would become the master node, coordinating the rest of the network) the special requirements of a WSN are hard to fulfill.
Being wireless, the nodes rely on batteries for their power supply. Therefore, it is important to keep the power consumption low. Compared to the other tasks of a WSN node, usage of the radio (to communicate) is very energy consuming. Therefore, if possible, the nodes should cooperate with as little communication as possible.

In the light of expensive communication, the methods above are not well suited for WSNs. For example, in the dual decomposition method, each subproblem depends on the vector of multipliers. Before every iteration, the node needs to update all of the multipliers. When the subproblem is solved, the node must send the residual back to the master problem. This creates a lot of message passing. What is worse, these methods often converge slowly. This means many iterations, and therefore lots of communication.

Fast-Lipschitz optimization is proposed with WSNs in mind. Typically, every node $i$ will be associated with one variable $x_i$ subject to one constraint $f_i$. The constraint can be a function of any set of variables in the network, but often only the ones of neighboring nodes. If the variables $x_i$ are collected in the vector $x$, this can be written

$$x_i \leq f_i(x)$$

for each node. If the components $f_i$ are collected in the vector valued function $f$, all the nodes can treated at once with

$$x \leq f(x).$$

The F-Lipschitz class consist of problems where the optimal point $x^*$ satisfies all constraints with equality,

$$x^* = f(x^*).$$

This eliminates the need for Lagrange multipliers since $x^*$ is found as the solution of a system of equations. In a distributed setup each node $i$ can update its own variable $x_i$ knowing only its own constraint function $f_i$ and the variables on which it depends. This is usually not all the variables of the network, but a few of the closest neighbors. As a consequence, the need for communication is lowered.

Intuitively, F-L optimization is fast. The reason is that the optimal point in some sense already is known. This can be seen as an optimization problem with only one feasible point.

The following chapter discusses F-L optimization in more detail.
7 Fast-Lipschitz Theory

This chapter concerns with the theory of F-L optimization.

We start this chapter by a formal definition of the F-L class in section 1, followed by the main theorem and its proof in section 2. Section 3 and 4 relates F-L to other classes of optimization problems and briefly talks about how to solve them. Finally, sections 5-7 discusses different forms of the original problem, and section 8 gives examples of F-L optimization.

Please note that this chapter builds heavily on ideas originally presented in [1]. This is especially true of sections 2 and 5. As it would be unpractical to everywhere point this out, or to treat the contributions of [1] and this thesis in separate chapters, the subject will be presented as a whole.

7.1 Definition

In this section, two variants of F-L problems are defined. The difference is whether the optimal point is guaranteed to be unique or not. This is discussed further in the next section.

**Definition 3** (Fast-Lipschitz optimization problem). A Fast-Lipschitz problem (F-L) has the form

\[
\begin{align*}
\max & \quad f_0(x) \\
\text{s.t.} & \quad x_i \leq f_i(x) \quad \forall i \in \mathcal{I} \\
& \quad x_i = f_i(x) \quad \forall i \in \mathcal{E} \\
& \quad x \in \mathcal{D} \subset \mathbb{R}^n
\end{align*}
\]

(9)

where \( \mathcal{I} \) and \( \mathcal{E} \) are complementary subsets of \( \{1, \ldots, n\} \), i.e.

\( \mathcal{I} \cup \mathcal{E} = \{1, \ldots, n\} \) and \( \mathcal{I} \cap \mathcal{E} = \emptyset \).

The last constraint is a box constraint, i.e. \( \mathcal{D} = \{x \in \mathbb{R}^n \mid a \leq x \leq b\} \).\(^6\) For \( i = 1, \ldots, n \), each \( f_i : \mathbb{R}^n \to \mathbb{R} \) is the \( i \)th component of the vector valued

\(^6\)The constraining set \( \mathcal{D} \) may often relate to physical limits of a control variable, wherefore the box constraint form comes naturally. It is only required however, that \( \mathcal{D} \) is convex and closed. For example, it could take on the form

\[ \mathcal{D} = \{x : ||x|| \leq 1\} \]

for some norm \( || \cdot || \).
function
\[ f(x) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{bmatrix}. \]

For all \( x \) in \( D \), the functions
\[ f_0 : D \subset \mathbb{R}^n \rightarrow \mathbb{R}^m, \quad m \geq 1 \]
and
\[ f : D \subset \mathbb{R}^n \rightarrow D \]
are continuously differentiable and fulfill conditions 4 and 5, given next.

Remark. It is possible that either \( I \) or \( E \) are empty, i.e. only equality constraints or only inequality constraints.

**Condition 4** (Fast-Lipschitz qualifying conditions). The following conditions should hold for all \( x \in D \):

\[
\begin{align*}
(0.a) \quad & \nabla f_0(x) \geq 0 & (10a) \\
(0.b) \quad & ||\nabla f(x)||_1 < 1 & (10b)
\end{align*}
\]

and

\[
\begin{align*}
(i.a) \quad & \nabla f(x) \geq 0 & (10c) \\
(ii.a) \quad & \nabla_i f_0(x) = \nabla_j f_0(x) \quad \forall i, j & (10d) \\
(ii.b) \quad & \nabla f(x) \leq 0 & (10e) \\
(ii.c) \quad & ||\nabla f(x)||_\infty < 1 & (10f)
\end{align*}
\]

or

\[
\begin{align*}
(iii.a) \quad & f_0(x) > 0 \in \mathbb{R} & (10g) \\
(iii.b) \quad & ||\nabla f(x)||_\infty < \frac{\delta}{\delta + \Delta} & (10h)
\end{align*}
\]

where \( \delta = \min_{i,x} \nabla_i f_0(x) \)
and \( \Delta = \max_{i,x} \nabla_i f_0(x) \)

or

\[
\begin{align*}
(iv.a) \quad & \nabla_i f_0(x) = \nabla_j f_0(x) \quad \forall i, j & (10i) \\
(iv.b) \quad & ||\nabla f(x)||_\infty < \frac{1}{2} & (10j)
\end{align*}
\]

Remark. The infinity-norm requirement of case (ii) can usually be relaxed, see section 7.2.2.
Condition 5 (Uniqueness condition). At the point $x^* = f(x^*)$, $\nabla f_0(x^*)$ contain no all-zero rows, i.e.

$$\nabla_i f_0(x^*) \neq 0^T \quad \forall i.$$ (11)

Remark. In general, the point $x^*$ is not known until the problem is solved. However, $x^*$ must be feasible wherefore $x^* \in D$ and condition (11) will certainly hold if $\nabla f_0(x)$ contain no all-zero rows for any $x \in D$.

Also note that case (iii) of condition 4 automatically fulfills (11).

In many application, the uniqueness of an optimal point is not crucial. One might be satisfied with finding any optimal point wherefore the next notion can be of use.

Definition 6. Weak Fast-Lipschitz
A weak F-L, is a problem of the form (9), fulfilling (only) condition 4.

To emphasize that a problem fulfills both conditions 4 and 5, it will sometimes be referred to as strong F-L (as opposed to weak F-L). “F-L” without further specification will always refer to the (strong) case of definition 3.

There is no difference in solving a weak F-L problem compared to a strong F-L. This means that if uniqueness is not required, one can simply solve the problem ignoring whether Condition (5) holds. One an optimal point is obtained, one can check if it also is unique.

7.2 Existence and uniqueness of solutions
We will now present and prove the main theorem of F-Lipschitz optimization. A proposition which allows for a significantly less cumbersome notation is also given.

Theorem 7. Existence and uniqueness of solution.
Let problem (9) be weak F-Lipschitz and feasible. Then, the problem has a Pareto optimal solution $x^*$, uniquely defined by

$$x_i^* = f_i(x^*) \quad \forall i = 1, \ldots, n.$$ 

Furthermore, if condition 5 applies (and problem (9) is strong F-L), then there are no other Pareto optimal points and $x^*$ is the unique maximizer.

Proof. Two alternative proofs are given shortly, in subsection 7.2.1. □

Uniqueness is always preferable though, since a non-singleton optimal set typically leaves the question “Which of optimal points is the best one?” (in some other sense then the objective of the original problem).
Remark. It can be seen in the proof that a problem qualifying through case (ii)-(iv) will always be (strongly) F-L.

In what follows, this next proposition allows for a simplified treatment and more compact notation.

**Proposition 8.** Let \( I_k \cap E_k = \emptyset \) and \( I_k \cup E_k = \{1, \ldots, n\} \) for \( k = 1, 2 \) and let the functions \( f_0 \) and \( f_i, i = 1, \ldots, n \) be shared among problems (P1) and (P2). Then, if

\[
\begin{align*}
(P1) & \quad \max_{x} f_0(x) \\
& \quad \text{s.t.} \quad x_i \leq f_i(x) \quad \forall i \in I_1 \\
& \quad x_i = f_i(x) \quad \forall i \in E_1 \\
& \quad x \in D
\end{align*}
\]

is F-L (with a unique optimum), so is

\[
\begin{align*}
(P2) & \quad \max_{x} f_0(x) \\
& \quad \text{s.t.} \quad x_i \leq f_i(x) \quad \forall i \in I_2 \\
& \quad x_i = f_i(x) \quad \forall i \in E_2 \\
& \quad x \in D
\end{align*}
\]

The two problems share the same set of solutions, the unique point

\[ x^* = f(x^*). \]

**Proof.** Changing from equality to inequality constraints does not affect the qualifying conditions. Therefore (P1) is F-L if and only if (P2) is F-L. By theorem 7, both problems share the unique optimal solution \( x^* \).

Remark. Although the proposition is true when \( I_1 = I_2 \), this case is useless. We are mostly interested in the case when \( I_2 = \emptyset \) and (P2) becomes

\[
\begin{align*}
\max_{x} f_0(x) \\
\text{s.t.} \quad x \leq f(x) \\
\quad x \in D
\end{align*}
\]

### 7.2.1 Proof of main theorem

The goal of this section is to prove theorem 7. Two proofs are presented. They look very similar, but rests on different foundations, each with its own merits.

First, an approach based on the mean value theorem is presented. This approach is intuitive and mostly uses basic calculus and linear algebra.

Thereafter, the foundations of the proof is instead formulated with the KKT conditions. This however requires the additional theory of *invex functions* to show that strong duality holds for problem (9), see [1] and the
Before we begin we need a few technical results.

**Lemma 9.** If problem (9) is F-L and \( \mathbf{x} \in \mathcal{D} \), then \( \mathbf{f}(\mathbf{x}) \) is a contraction with respect to the max-norm \( \| \cdot \|_\infty \).

**Proof.** The function \( \mathbf{f}(\cdot) \) is a contraction with respect to the max-norm on \( \mathcal{D} \) if it holds, for any \( \mathbf{x}, \mathbf{y} \in \mathcal{D} \) that

\[
\| \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}) \|_\infty < \| \mathbf{x} - \mathbf{y} \|_\infty. \tag{12}
\]

By the max-norm definition, \( \| \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}) \|_\infty = \max_j |f_j(\mathbf{x}) - f_j(\mathbf{y})| \) for all \( i \). Applying the mean value theorem to the \( j \)th component of \( \mathbf{f} \) gives

\[f_j(\mathbf{x}) - f_j(\mathbf{y}) = \nabla f_j(\mathbf{z}_j)^T (\mathbf{x} - \mathbf{y})\]

for some \( \mathbf{z}_j \) on the line between \( \mathbf{x} \) and \( \mathbf{y} \). Taking the absolute value of both sides and applying Hölder’s inequality gives

\[|f_j(\mathbf{x}) - f_j(\mathbf{y})| \leq \| \nabla f_j(\mathbf{z}_j) \|_1 \| \mathbf{x} - \mathbf{y} \|_\infty,\]

wherefore

\[\| \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}) \|_\infty = \max_j |f_j(\mathbf{x}) - f_j(\mathbf{y})| \leq \max_j \| \nabla f_j(\mathbf{z}_j) \|_1 \| \mathbf{x} - \mathbf{y} \|_\infty.\]

Finally,

\[
\max_j \| \nabla f_j(\mathbf{z}_j) \|_1 = \max_j \sum_i \nabla_i f_j(\mathbf{z}_j) = \| \nabla \mathbf{f}(\mathbf{z}_j) \|_1.
\]

Note that \( \nabla \mathbf{f}_j \) is a vector and \( \nabla \mathbf{f} \) is a matrix, wherefore the first \( \| \cdot \|_1 \) in the equation above represents a vector norm while the second represents an induced matrix norm. Qualifying property (0.b) now gives \( \| \nabla \mathbf{f}(\mathbf{z}_j) \|_1 < 1 \) for all \( \mathbf{z}_j \in \mathcal{D} \) (\( \mathbf{z}_j \in \mathcal{D} \) since \( \mathbf{x}, \mathbf{y} \in \mathcal{D} \) and \( \mathcal{D} \) is convex) wherefore \( \| \mathbf{f}(\mathbf{x}) - \mathbf{f}(\mathbf{y}) \|_\infty < \| \mathbf{x} - \mathbf{y} \|_\infty. \)

**Remark.**

**Lemma 10.** Let

\[\mathbf{A} = \int_0^1 \nabla \mathbf{f}(\mathbf{x}(t)) dt\]

for some curve \( \mathbf{x}(t) \) entirely within \( \mathcal{D} \). Then, the norms \( \| \mathbf{A} \|_1 \) and \( \| \mathbf{A} \|_\infty \) are bounded by the corresponding norms \( \| \nabla \mathbf{f} \|_{1/\infty} = \max_{\mathbf{x} \in \mathcal{D}} \| \nabla \mathbf{f} \|_{1/\infty} \).
Proof. Write

\[ ||A||_1 = \max_j \sum_i |\int_0^1 \nabla_i f_j(x(t)) dt| \leq \max_j \sum_i \int_0^1 |\nabla_i f_j(x(t))| dt \]
\[ \leq \max_j \sum_i \max_{t \in [0,1]} |\nabla_i f_j(x(t))| \cdot 1 \]
\[ \leq \max_{x \in D} \max_j \sum_i |\nabla_i f_j(x)| = \max_{x \in D} ||\nabla f(x)||_1 = ||\nabla f||_1. \]

By an identical argument,

\[ ||A||_\infty \leq ||\nabla f||_\infty. \]

\[ \square \]

**Proof based on integration** The idea of the proof is to show that there exists a unique point \( x^* \) such that \( x^* = f(x^*) \) and that no other feasible point \( y \) has a higher scalarized cost than \( x^* \), i.e. \( \mu^T f_0(y) < \mu^T f_0(x^*) \) for all feasible \( y \) and all \( \mu > 0 \) in \( \mathbb{R}^m \).

By lemma 9, \( f(x) \) is a contraction on \( D \). Since \( D \) is closed and \( f : D \to D \), Proposition 2 guarantees the existence of a fixed point \( x^* = f(x^*) \). The proposition further states that the fixed point \( x^* \) is unique.

Define \( \Psi(x) = x - f(x) \). A point \( x \) is only feasible if \( \Psi(x) \leq 0 \) and the definition of \( x^* \) gives \( \Psi(x^*) = 0 \). Pick an arbitrary feasible point \( y \neq x^* \) and denote the difference \( (x^* - y) = r \). Because \( y \) is feasible, \( \Psi(y) = -b \) where \( b \geq 0 \) and \( b \neq 0 \) (if \( b = 0 \) then \( y = f(y) \) but \( x^* \) is the unique fixed point of \( f \)). Every feasible point, in particular \( x^* \) and \( y \), is also required to lie within \( D \).

For these point \( y \) and \( x^* \) we have

\[ b = \Psi(x^*) - \Psi(y) = \int_0^1 d\Psi(y + tr) dt = \int_0^1 J\Psi(y + tr) dt = \left( I - \int_0^1 J f(y + tr) dt \right) r. \]  

(13)

The point \( x(t) = y + tr = (1 - t) y + tx^* \) need not be feasible but since \( D \) is convex, \( x(t) \) will always lie within \( D \) (because \( x^*, y \in D \) and \( t \in [0,1] \)).

To simplify notation, define the matrix \( A \) as

\[ A = \int_0^1 Jf_i(x(t)) dt = \int_0^1 \nabla f(x(t)) dt. \]  

(14)
This gives
\[ b = \begin{pmatrix} I - A^T \end{pmatrix} r \]
and
\[ r^T = b^T (I - A)^{-1}. \]  \hspace{1cm} (15)

The inverse exists since the eigenvalues, \( \text{eig}_i(I - A) = 1 - \text{eig}_i(A) \), are strictly positive. This follows from lemma 10 and part (0.b) of the qualifying constraints:
\[ \rho(A) \leq ||A||_1 \leq ||\nabla f(x)||_1 < 1 \]
wherefore
\[ \min_i (1 - \text{eig}_i(A)) = 1 - \max_i \text{eig}_i(A) \geq 1 - \rho(A) > 0. \]

Denote the difference in the scalarized cost function between \( x^* \) and \( y \)
\[ \theta = \mu^T f_0(x^*) - \mu^T f_0(y) = \mu^T \int_0^1 J f_0(y + tr) r dt = c^T r, \]  \hspace{1cm} (16)
where
\[ c = \int_0^1 J f_0^T(y + tr) dt \mu = \int_0^1 \nabla f_0(x(t)) dt \mu \]  \hspace{1cm} (17)
is a vector in \( \mathbb{R}^n \). Once again, \( x(t) = y + tr \in \mathcal{D} \). \( c \) can be seen as the average slope of \( f_0 \) on the line segment from \( y \) to \( x^* \).

Qualifying condition (0.a) states \( \nabla f_0(x) \geq 0 \ \forall x \in \mathcal{D} \). Therefore, the integral part of \( c \) (being the integral of a non-negative integrand) is always non-negative. Therefore, since \( \mu > 0 \), it will always hold that
\[ c \geq 0. \]

If condition 5 (uniqueness) also holds at least one component of each row of \( \nabla f_0(x^*) \) is positive. This component will also be positive in some neighborhood of \( x^* \), since \( f_0 \) is continuously differentiable. For any \( t = 1 - \epsilon \) with \( \epsilon > 0 \) small enough, \( x(t) \) lies in this neighborhood and
\[ \int_0^1 \nabla f_0(x(t)) dt \mu = \int_0^{1-\epsilon} \nabla f_0(x(t)) dt \mu + \int_{1-\epsilon}^1 \nabla f_0(x(t)) dt \mu. \]
The first term is still non-negative. The second term is now positive, since \( \int_{1-\epsilon}^1 \nabla f_0(x(t)) dt \mu \) has at least one positive component per row and \( \mu > 0 \). Therefore (when condition 5 holds)
\[ c > 0. \]

If the difference \( \theta \) in (16) is non-negative for every \( r \), then no other point \( y \) has a higher cost function value than \( x^* \), i.e. \( x^* \) is optimal. If \( \theta \) is also
strictly positive, then $x^*$ has a higher cost than any other feasible point $y$ and $x^*$ is the unique optimum.

Since $\theta$ is scalar, it is equal to its transpose and can be rewritten (with equation (15)) as

$$\theta = c^T r = r^T c = b^T (I - A)^{-1} c = b^T \lambda.$$  

As $b$ depends on the arbitrary point $y$, all we can say about it is $b \geq 0$ and $b \neq 0$ (since $y$ is feasible). This can be said regardless of the properties of $f$ and $f_0$, whose effects are collected in the new variable $\lambda$.

As $\lambda$ is scalar, it is equal to its transpose and can be rewritten (with equation (15)) as

$$\lambda = (I - A)^{-1} c.$$  

Clearly, $\theta \geq 0$ if $\lambda \geq 0$ and $\theta > 0$ if $\lambda > 0$.

The rest of this proof will, for each case of (10), show that $\lambda \geq 0$ when $c \geq 0$ and $\lambda > 0$ when $c > 0$.

**Case (i)** We will use $\lambda = (I - A)^{-1} c$ as it stands. As $\rho(A) < 1$, the inverse has a converging Taylor expansion (Proposition 1)

$$(I - A)^{-1} = \sum_{k=0}^{\infty} A^k.$$  

From the definition of $A$ in (14) it is clear that $A \geq 0$ when part (i) of Condition 4 hold ($\nabla f \geq 0$). Therefore, by (19), $(I - A)^{-1} \geq 0$. This together with $c \geq 0$ gives

$$\lambda = (I - A)^{-1} c \geq 0.$$  

Since $(I - A)^{-1}$ is invertible it has full rank, i.e. every row has at least one non-zero component. Therefore, $c > 0$ guarantees $\lambda > 0$.

**Case (ii)** Rewriting (18) gives $\lambda = c + A\lambda$ and by iterating this we get

$$\lambda = c + A\lambda = c + A(c + A\lambda) = (I + A)c + A^2 \lambda,$$

or

$$\lambda = (I - A^2)^{-1} (I + A)c.$$  

---

8 $\lambda$ is closely connected to the Lagrange multipliers of the problem, see the “Lagrangian proof”

9 See section 3.4.
As in case (i) $\rho(A^2) = \rho(A)^2 < 1$ and the inverted matrix can be written

$$(I - A^2)^{-1} = \sum_{k=0}^{\infty} (A^2)^k$$

which, as seen in (14), is guaranteed non-negative when $\nabla f(x) \leq 0$ (condition (ii.b)).

When condition (ii.a) hold the columns of $\nabla f_0$ are identical whereby any vector $\nabla f_0 \mathbf{v}$ is parallel to $1$. Therefore, for any $\mathbf{\mu} > 0$, (17) gives $\mathbf{c} = c \mathbf{1}$ for some scalar $c > 0$. Hence,

$$(I + A)c = c(I + A)1$$

and as required by condition (ii.c), $||A||_\infty < 1$ so the minimal element

$$\min_{i} \sum_{j=1}^{n} (I + A)_{ij}1_j = 1 - \max_{i} \sum_{j=1}^{n} A_{ij} = 1 - ||A||_\infty$$

is always positive, i.e $(I - A)c > 0$. Therefore, since $(I - A^2)^{-1}$ is invertible and has non-zero rows, 10

$$\lambda = (I - A^2)^{-1}(I + A)c > 0.$$ 

**Case (iii)** In this case $\nabla f_0(x)$ is scalar, which means $\mathbf{\mu}$ just scales the cost and can be omitted. $\nabla f_0(x) > 0$ gives $\mathbf{c} > 0$.

Write (18) as

$$\lambda = A\lambda + \mathbf{c}$$

and consider the $i^{th}$ row

$$\lambda_i = \sum_{j=1}^{n} A_{ij} \lambda_j + c_i.$$ 

By taking the absolute value of this equation, one can bound the magnitude of $\lambda_i$,

$$|\lambda_i| \leq \left| \sum_{j=1}^{n} A_{ij} \lambda_j \right| + |c_i| \leq \sum_{j=1}^{n} |A_{ij}||\lambda_j| + c_i \quad \forall i.$$ 

Introducing $\lambda_{\max} = \max_i |\lambda_i|$ we get

$$\lambda_{\max} \leq \sum_{j=1}^{n} |A_{ij}|\lambda_{\max} + c_i \leq \max_i \sum_{j=1}^{n} |A_{ij}|\lambda_{\max} + \max_i c_i.$$ 

\[10\] See section 3.4.
Noting that max \( \sum_{j=1}^{n} |A_{ij}| = ||A||_\infty \) and denoting max \( c_i \) with \( c_{\text{max}} \) we get (assuming \( ||A||_\infty < 1 \))

\[
\lambda_{\text{max}} \leq \frac{c_{\text{max}}}{1 - ||A||_\infty}.
\]

Similarly, a lower bound on \( \lambda_i \) can be obtained from

\[
\lambda_i \geq - \sum_{j=1}^{n} |A_{ij}| \lambda_j + c_i \geq - \sum_{j=1}^{n} |A_{ij}| \lambda_j + c_i \geq - \sum_{j=1}^{n} |A_{ij}| \lambda_{\text{max}} + c_i \quad \forall i.
\]

Introducing \( \lambda_{\text{min}} = \min_i \lambda_i \) (no absolute value) and \( c_{\text{min}} = \min_i c_i \) we get

\[
\lambda_{\text{min}} \geq \min_i \left( - \sum_{j=1}^{n} |A_{ij}| \lambda_{\text{max}} + c_i \right)
\]

\[
\geq - \max_i \sum_{j=1}^{n} |A_{ij}| \lambda_{\text{max}} + c_{\text{min}} = - ||A||_\infty \lambda_{\text{max}} + c_{\text{min}}.
\]

Inserting the bound on \( \lambda_{\text{max}} \) into the expression above gives

\[
\lambda_{\text{min}} \geq - ||A||_\infty \frac{c_{\text{max}}}{1 - ||A||_\infty} + c_{\text{min}}.
\]

It is already assumed that \( ||A||_\infty < 1 \), so \( (1 - ||A||_\infty) > 0 \) and

\[
\lambda_{\text{min}} \geq \frac{c_{\text{min}} - ||A||_\infty (c_{\text{min}} + c_{\text{max}})}{1 - ||A||_\infty}.
\]

(20)

The right hand side above has a positive denominator and the numerator is positive when

\[
c_{\text{min}} - ||A||_\infty (c_{\text{max}} + c_{\text{min}}) > 0 \Leftrightarrow ||A||_\infty < \frac{c_{\text{min}}}{c_{\text{max}} + c_{\text{min}}}
\]

(recall \( c > 0 \Rightarrow c_{\text{min}}, c_{\text{max}} > 0 \)). When qualifying constraint (iii.b) holds \( (||\nabla f(x)||_\infty < \frac{\delta}{\delta + \Delta}) \) Lemma 10 gives

\[
||A||_\infty \leq ||\nabla f(x)||_\infty < \frac{\delta}{\Delta + \delta} < \frac{c_{\text{min}}}{c_{\text{max}} + c_{\text{min}}} < 1,
\]

since

\[
c_{\text{min}} = \min_i \int_{0}^{1} \nabla_i f_0(y + tr) dt \geq \min_{x_i} \nabla_i f_0(x) = \delta
\]

and

\[
c_{\text{max}} = \max_i \int_{0}^{1} \nabla_i f_0(y + tr) dt \leq \max_{x_i} \nabla_i f_0(x) = \Delta.
\]
Therefore, the assumption \( \|A\|_\infty < 1 \) holds and the right hand side of (20) is positive,
\[
\lambda_{\text{min}} \geq \frac{c_{\text{min}} - \|A\|_\infty (c_{\text{min}} + c_{\text{max}})}{1 - \|A\|_\infty} > 0.
\]
As \( \lambda_i \geq \lambda_{\text{min}} \) for all \( i \), \( \lambda > 0 \) and so
\[
r^T c = b^T (A - I)^{-1} c = b^T \lambda > 0.
\]

Case (iv)  In this case, we combine the techniques from case (i) and (ii). As in case (ii), \( \nabla_i f_0 = \nabla_j f_0 \) (condition iv.a) gives \( c = c1 \). Expanding \( \lambda \) as in case (i) (still allowed since \( \rho(A) \leq \|\nabla f(x)\|_\infty < 1/2 \)) gives
\[
\lambda = (I - A)^{-1} c = (I + A + A^2 + \ldots) c
\]
\[
= c \left( I + A \sum_{k=0}^{\infty} A^k \right) 1.
\]
As in case (ii), the \( i^{th} \) component is
\[
\lambda_i = c \left( 1 + \sum_{j} \left[ A \sum_{k=0}^{\infty} A^k \right]_{ij} \right) \geq c \left( 1 - \|A\|_\infty \sum_{k=0}^{\infty} \|A^k\|_\infty \right)
\]
and \( c > 0 \) gives \( \lambda > 0 \) if \( \|A\|_\infty < 1 \). Using the basic norm inequalities and the sum of the infinite geometric series we get
\[
\|A\|_\infty \sum_{k=0}^{\infty} \|A^k\|_\infty \leq \|A\|_\infty \cdot \sum_{k=0}^{\infty} \|A\|_\infty^k = \frac{\|A\|_\infty}{1 - \|A\|_\infty} < 1
\]
The geometric series converges when \( \|A\|_\infty < 1 \) and the fraction is less than one when \( \|A\|_\infty < 1/2 \). Condition (iv.b) requires \( \|f\|_\infty < 1/2 \), wherefore \( \|A\|_\infty < \|f\|_\infty < 1/2 < 1 \) and so
\[
\|A\|_\infty \sum_{k=0}^{\infty} \|A^k\|_\infty < 1 \Rightarrow \lambda > 0.
\]
\[
\square
\]

Proof based on KKT conditions

47
This section proves nothing not already proved by the previous proof. In fact, it is weaker in the sense that it can only prove the strong version of F-L. However, it is still interesting for several reasons. For example, it ties in to duality theory and might offer new possibilities to generalize the theory. Consider problem (9) (scalarized, $\mu > 0$) with inequality constraints only:

$$\max \quad \mu^T f_0(x)$$
$$\text{s.t.} \quad x_i \leq f_i(x) \quad 1, \ldots, n$$

By proposition 8, this does not result in loss of generality. The Lagrangian is

$$L(x, \lambda) = \mu^T f_0(x) - \lambda^T (x - f(x)).$$

Assuming that strong duality holds, any pair $(\hat{x}, \hat{\lambda})$ of candidates for optimal variables must satisfy the KKT-conditions (see the section 5.4.4). In particular, $\hat{x}$ must be the minimizer of $L(x, \hat{\lambda})$ which requires

$$\nabla_x L(\hat{x}, \hat{\lambda}) = 0$$

and strict complementarity must hold, i.e.,

$$\hat{\lambda}^T (\hat{x} - f(\hat{x})) = 0.$$  (22)

The idea of the proof is showing that $\hat{\lambda}_i > 0$ for all $i$, wherefore (22) requires $(\hat{x}_i - f_i(\hat{x})) = 0$ for all $i$. As in the previous proof, $f(x)$ is a contraction on $\mathcal{D}$ wherefore the unique optimal point is $\hat{x} = x^* = f(x^*)$.

From (21) we have

$$\nabla_x L(\hat{x}, \hat{\lambda}) = \nabla f_0(\hat{x}) \mu - \hat{\lambda} + \nabla f(\hat{x}) \hat{\lambda} = 0.$$  (23)

For a simpler notation we introduce

$$c = \nabla f_0(\hat{x}) \mu \quad \text{and} \quad A = \nabla f(\hat{x}).$$

Note that these notations are similar (but not identical) to those of of the previous proof.

Condition (23) can now be written

$$\hat{\lambda} = A \hat{\lambda} + c, \quad \text{or} \quad \hat{\lambda} = (I - A)^{-1} c.$$  (24)

The form of this equation is identical to (18) of section 7.2.1. To complete the proof, one must show that $A$ and $c$ have the same properties as their respective section 7.2.1 counterparts. Then, by the exact argument of the

11 Any candidate for optimality must also be feasible, wherefore $\hat{x} \in \mathcal{D}$.  

48
first proof, $\lambda > 0$.

As $\hat{x} \in D$, the properties of $A = \nabla f(\hat{x})$ and $c = \nabla f_0(\hat{x})\mu$ are given almost directly by the qualifying conditions. Verifying can be done in a case-by-case basis, but will not be presented here.

\[\Box\]

7.2.2 Relaxation of max-norm conditions

As of now there is only a result for case (ii). The cases (iii) and (iv) can be shown similarly and should be easier in a sense that they do not impose sign restrictions on $\nabla f$ or $\nabla g$. This is left for future investigation.

Case (ii) Case (ii) of the qualifying conditions require, besides $||\nabla f||_\infty < 1$, $\nabla f \leq 0$. Instead of verifying these constraints for $\nabla f$, we will transform $f(x)$ into an equivalent constraint function $g(x)$. For simplified notation, consider problem (9) with inequality constraints only:

\[
\begin{align*}
\max & \quad f_0(x) \\
\text{s.t.} & \quad x \leq f(x) \\
& \quad x \in D
\end{align*}
\]

(25)

By Proposition 8, this can be done without loss of generality. For all $i = 1, \ldots, n$, let

\[g_i(x) = x_i - \gamma_i (x_i - f_i(x))\]

with $\gamma_i > 0$. On vector form, this can be written

\[g(x) = x - \Gamma (x - f(x))\]

with $\Gamma = \text{diag}(\gamma)$ for a constant vector $\gamma > 0$. Formulate the new problem

\[
\begin{align*}
\max & \quad f_0(x) \\
\text{s.t.} & \quad x \leq g(x) \\
& \quad x \in D.
\end{align*}
\]

(26)

The problems (25) and (26) are equivalent since $x_i \leq g_i(x)$ if and only if $\gamma_i (x_i - f_i(x)) \leq 0$, i.e., $x_i \leq f_i(x)$. We are still aiming to fulfill case (ii), wherefore we need $\nabla g \leq 0$ and $||\nabla g||_\infty < 1$.

Assume $\nabla f \leq 0$ already holds. The components of $\nabla g$ are

\[\nabla g_j(x) = \delta_{ij} - \gamma_i (\delta_{ij} - \nabla f_j(x))\]

(27)

and $\nabla g \leq 0$ only if they fulfill
\[
\begin{align*}
\gamma_j \nabla_i f_j (x) &\leq 0 \quad i \neq j \\
1 - \gamma_i (1 - \nabla_i f_i (x)) &\leq 0 \quad i = j.
\end{align*}
\]

The off diagonal case is true by assumption if \( \gamma > 0 \). The diagonal case is true only when
\[
\gamma_i \geq \frac{1}{1 - \nabla_i f_i (x)} \quad \forall x \in D.
\]

(28)

\( \nabla f \leq 0 \) guarantees the denominator above is always positive. Introduce \( \gamma_{\text{min}} \) as the smallest constant vector where each component fulfills (28),
\[
[\gamma_{\text{min}}]_i = \max_{x \in D} \frac{1}{1 - \nabla_i f_i (x)} = \frac{1}{1 - \max_{x \in D} \nabla_i f_i (x)}.
\]

We want \( ||\nabla g||_{\infty} < 1 \). By the definition of the infinity-norm this requires
\[
||\nabla g||_{\infty} = \max_i \left\{ |1 - \gamma_i (1 - \nabla_i f_i (x))| + \sum_{j \neq i} |\gamma_j \nabla_i f_j (x)| \right\}
\]
\[
= \max_i \left\{ -1 + \gamma_i (1 - \nabla_i f_i (x)) + \sum_{j \neq i} |\nabla_i f_j (x)| \gamma_j \right\} < 1.
\]

This is true if and only if
\[
(1 - \nabla_i f_i (x)) \gamma_i + \sum_{j \neq i} |\nabla_i f_j (x)| \gamma_j < (1 + 1) \quad \forall i \text{ and } x \in D.
\]

Since \( \nabla_i f_i \leq 0 \Rightarrow -\nabla_i f_i \gamma_i = |\nabla_i f_i| \gamma_i \) we can include this term in the sum. The condition above now becomes
\[
\gamma_i + \sum_j |\nabla_i f_j (x)| \gamma_j < 2 \quad \forall i \text{ and } x \in D,
\]

(29)

or in vector form
\[
M \gamma < c
\]

(30)

where \( M = (I + |\nabla f|) \geq 0 \) and \( c = 2 \cdot 1 \).\(^{12}\)

Any permissible \( \gamma \) can be written \( \gamma = \gamma_{\text{min}} + \epsilon \) with a \( \epsilon \geq 0 \) not too large. Therefore (since \( M \geq 0 \)),
\[
M \gamma = M (\gamma_{\text{min}} + \epsilon) \geq M \gamma_{\text{min}}
\]

\(^{12}\)The inverse of \( M \) is non-negative if and only if \( M \) is a monomial matrix (generalized permutation matrix), i.e. \( M \) has exactly one non-zero entry in each row and each column. Since the diagonal elements of \( M \) are positive, \( M^{-1} \geq 0 \) if and only if \( M \) is diagonal. [14, p. 68]
and \( M\gamma < c \) if and only if \( M\gamma_{\text{min}} < c \).

Introducing \( \gamma_{\text{min}} \) in (29) gives

\[
\frac{1}{1 - \max_{x \in D} \nabla_i f_i(x)} + \sum_j \frac{|\nabla_i f_j(x)|}{1 - \max_{x \in D} \nabla_j f_j(x)} < 2 \quad \forall i \text{ and } x \in D
\]

which is hard to verify, in a general case.

One could argue that

\[
\nabla f \leq 0 \Rightarrow 1 - \max_{x \in D} \nabla_i f_i(x) \leq 1,
\]

but this will result in \( 1 - \|\nabla f\|_\infty < 2 \) which is exactly the condition we want to relax. From this attempt however it is clear that in order to relax the condition, we want large magnitudes for the diagonal elements of \( \nabla f \), i.e. the diagonal elements of \( \nabla f \) should be as negative as possible.

Introduce \( s_k = |\max_{x \in D} \nabla_k f_k(x)| \) and let \( s \) be the diagonal element of \( \nabla f \) with the smallest magnitude over all \( x \in D \), i.e.

\[
s = \min_k s_k.
\]

Note that \( s \geq 0 \). Another way to put it is that \(-s\) is the least negative diagonal element of \( \nabla f \) over all \( x \in D \). Now, choosing \( \gamma = \left( \frac{1}{1+s} \right) 1 \) ensures \( \gamma \geq \gamma_{\text{min}} \) since

\[
[\gamma_{\text{min}}]_i = \frac{1}{1 - \max_{x \in D} \nabla_i f_i(x)} \leq \frac{1}{1+s}.
\]

Inserting this \( \gamma \) in (29) gives the condition

\[
\frac{1 + \sum_j |\nabla_i f_j(x)|}{1 + s} < 2 \quad \forall i \text{ and } x \in D
\]

which is equivalent to

\[
\frac{1 + \|\nabla f\|_\infty}{1 + s} < 2,
\]

or \( \|\nabla f\|_\infty < 2(1 + s) - 1 = 2s + 1 \).

This is indeed more generous than \( \|\nabla f\|_\infty < 1 \), as long as the diagonal elements of \( \nabla f \) are strictly negative. The more diagonally dominant \( \nabla f \) is, the better.

Unfortunately, any zero diagonal element (at any point \( x \in D \)) will render this technique useless. Transforming the constraint function as follows might help.
Pre-transform  This transform aims to increase the magnitudes of the diagonal elements of $\nabla f$.
Consider the $j^{th}$ constraint,

$$x_j \leq f_j(x),$$  \hfill{(32)}

and remove $a_j x_j$ from each side

$$x_j - d_j x_j \leq f_j(x) - d_j x_j.$$  

This can be written as

$$x_j \leq \frac{f_j(x) - d_j x_j}{1 - d_j} = h_j(x).$$  \hfill{(33)}

Then, $x_j \leq f_j(x)$ and $x_j \leq h_j(x)$ are equivalent if $d_j < 1$. Further, $\nabla_j h_j$ is more negative (has a larger magnitude) than $\nabla_j f_j$ if $d_j > 0$. Unfortunately there is also a downside to this transform. When $d_j$ increases towards 1, every element of $|\nabla h_j|$ approaches infinity, i.e., $||\nabla h||_\infty$ will eventually grow to large.

Applying the transform to all constraints at once can be accomplished with the matrix $D = \text{diag}(d_1, \ldots, d_n)$,

$$x \leq h(x) = (I - D)^{-1} (f(x) - Dx).$$  \hfill{(34)}

This gives

$$\nabla h(x) = \nabla (f(x) - Dx) (I - D)^{-T} = (\nabla f(x) - D^T) (I - D)^{-T} = (\nabla f(x) - D) (I - D)^{-1} = \nabla f(x) (I - D)^{-1} - D (I - D)^{-1}.$$  

The infinity norm of the transformed constraints is

$$||\nabla h||_\infty = \max_i ||\nabla_i h||_1 = \max_x \max_i \sum_j \left| \frac{\nabla_i f_j(x)}{1 - d_j} - \delta_{ij} \frac{d_i}{1 - d_i} \right|_{\leq 0} \sum_j \left| \frac{\nabla_i f_j(x)}{1 - d_j} + \delta_{ij} \frac{d_i}{1 - d_i} \right|_{\geq 0}$$  

$$= \max_i \left\{ \max_x \sum_j \left| \frac{\nabla_i f_j(x)}{1 - d_j} + \delta_{ij} \frac{d_i}{1 - d_i} \right| \right\},$$  \hfill{(35)}

and

$$s_k = \max_{x \in D} \nabla_k h_k(x) = \max_{x \in D} \nabla_k f_k(x) - d_k = -\max_{x \in D} \nabla_k f_k(x) - d_k.$$  \hfill{(36)}

Both $||\nabla h||_\infty$ and $s$ are increasing functions of $D$ (remember $0 \leq d_k < 1$ and $\nabla f \leq 0$).
Replacing $\nabla f$ with $\nabla h$ in (31) gives the new requirement

$$\frac{1 + \|\nabla h\|_\infty}{1 + s} = R(D) < 2.$$ 

Inserting $s$ and $\|\nabla h\|_\infty$ into this expression gives the following result:

**Proposition 11.** When $\nabla f \leq 0$, then a set of equivalent constraints $\tilde{f}(x)$ satisfy $\nabla \tilde{f} \leq 0$ and $\|\nabla \tilde{f}\|_\infty < 1$ if there is some diagonal matrix $0 \leq D < I$ such that

$$R(D) = \frac{1 + \max_i \left\{ \max_x \sum_j \left| \frac{\nabla f_i(x)}{1 - d_j} \right| + \frac{d_i}{1 - d_i} \right\}}{1 - \min_k \max_x \frac{\nabla f_k(x) - d_k}{1 - d_k}} < 2. \quad (37)$$

It remains to determine the $D$ ($0 \leq D < I$) minimizing $R(D)$. This problem is yet to be solved, and is left for future investigation. Due to its combinatorial nature it might be hard to find an analytical solution, but some hints of the solution (or at least an algorithm for solving it) can be found by inspection. For example, let $i^*$ and $k^*$ be the indexes currently optimizing the sub-problems of the last line of (37). Starting from $D = 0$ (no transformation) it makes no sense to increase a $d_k$ with $k \neq k^*$ since that would just increase $\|\nabla g\|_\infty$ and leave $s$ unaffected, thus increasing $R$.

### 7.3 Fast-Lipschitz and other classes of optimization problems

Unlike many of the frameworks for optimization existing today, F-L problems does not need to be convex. Still, some convex problems, such as linear, quadratic and geometric problems can be rewritten and solved with F-L theory (see Section 7.5).

Interference function theory, important for optimizing radio powers in cellular networks, can be shown to be a special case of F-L optimization [1, 13].

### 7.4 Solving a Fast-Lipschitz problem

The great benefit of F-L problems is the fact that, once classified as F-L, their optimal point is implicitly known. All one needs to is solve the $n$ by $n$ system of equations,

$$x^* = f(x^*) \quad (38)$$
In general, this is much easier than solving an optimization problem in \( n \) variables, which often requires solving systems of twice the size (one dual variable for each constraint).

If the problem (in practice) admits a centralized solution, any suitable method for non-linear equation solving can be used. Fast alternatives include Newton-like methods, but which particular method to use (and tuning parameters) vary depending on the specific form and parameters of the problem.

The real strength of F-L theory however, is the ability to solve (38) in a distributed way. As discussed in the introduction, this is often required in WSN application.

A simple method uses the fact that \( f(x) \) is a contraction (by proposition 9) with \( x^* \) as its fixed point. With this method, \( x^* \) is found as the limit of the sequence \( x^{k+1} = f(x^k) \). This results can be found in the original paper on F-L theory \[1\].

If the iterations are synchronized, every node \( i \) computes \( x^{k+1}_i = f_i(x^k) \) at time \( k \). The vector \( x \) will then evolve as \( ||x^k - x^*||_\infty \leq \alpha^k ||x^0 - x^*||_\infty \), where the modulus \( \alpha = ||\nabla f||_1 < 1 \). It is possible to bound the number of iterations needed for a required accuracy of the solution as follows. Denote the required accuracy \( \epsilon \). After at most \( \tilde{k} \) iterations we want \( ||x^{\tilde{k}} - x^*||_\infty \leq \epsilon \) or

\[
||x^{\tilde{k}} - x^*||_\infty \leq \alpha^{\tilde{k}} ||x^0 - x^*||_\infty \leq \epsilon 
\] (39)

The initial distance (if we define the distance function as \( d(x, y) = ||x - y||_\infty \)) from optimality can be bounded by \( d \), the largest distance between two points in \( D \). As \( D \) is box shaped, this means \( ||x^0 - x^*||_\infty < d = ||b - a||_\infty \). Now, (39) is true if

\[
\alpha^{\tilde{k}} \leq \frac{\epsilon}{||x^0 - x^*||_\infty} \leq \frac{\epsilon}{d}
\]

and, taking the logarithm of both sides, \( \tilde{k} \ln \alpha \leq \ln \epsilon/d \). Both \( \alpha \) and \( \epsilon/d \) are less than one, wherefore \( \ln \alpha \) and \( \ln \epsilon/d \) are negative and the requirement on \( \tilde{k} \) becomes

\[
\tilde{k} \geq \frac{\ln \epsilon/d}{\ln \alpha} = \frac{\ln \epsilon - \ln d}{\ln \alpha}
\]

The method can also be shown to converge in a completely asynchronous setting, although there are no simple results like the one above bounding the number of iterations. See \[1\] for the technical result.
7.5 Canonical form

The Fast-Lipschitz formulation (9) differs from the canonical form, which is the standard way of formulating an optimization problem. The canonical form is given by

\[
\begin{align*}
\min & \quad g_0(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \ i = 1, \ldots, l \\
& \quad g_i(x) = 0, \ l + 1, \ldots, n \\
& \quad x \in D = \{x \in \mathbb{R}^n : a \leq x \leq b\}
\end{align*}
\]

where

\[
\begin{align*}
g_0(x) : D \to \mathbb{R}^m & \quad m \geq 1 \\
g_i(x) : D \to \mathbb{R} & \quad i = 1, \ldots, n
\end{align*}
\]

It is interesting to see when a problem in canonical form qualifies as F-Lipschitz problem. Transform problem (40) by introducing

\[
\begin{align*}
f_0(x) &= -g_0(x) \\
f_i(x) &= x_i - \gamma_i g_i(x) & \quad i = 1, \ldots, n
\end{align*}
\]

and consider the problem

\[
\begin{align*}
\max & \quad f_0(x) \\
\text{s.t.} & \quad x_i \leq f_i(x), \ i = 1, \ldots, l \\
& \quad x_i = f_i(x), \ l + 1, \ldots, n \\
& \quad x \in D
\end{align*}
\]

which is on FL form. Minimizing \(g_0\) is the same as minimizing \(-g_0 = f_0\), so the objectives of (40) and (42) are the same. If the constants \(\gamma_i\) are restricted to

\[
\begin{align*}
\gamma_i > 0 & \quad i = 1, \ldots, l \\
\gamma_i \neq 0 & \quad i = l + 1, \ldots, n
\end{align*}
\]

problem (40) and (42) also have equivalent constraints, i.e.

\[
\begin{align*}
g_i(x) \leq 0 & \iff x_i \leq f_i(x), \ i = 1, \ldots, l \\
g_i(x) = 0 & \iff x_i = f_i(x), \ i = l + 1, \ldots, n
\end{align*}
\]

What requirements on problem (40) ensures that (42) fulfills the qualifying conditions (10)?

The following theorem is given in [1] and will not be proved here.

**Theorem 12.** Let \(g(x) = [g_1(x), \ldots, g_n(x)]^T\). The following conditions guarantee problem (40) can be re-written and solved as a Fast-Lipschitz problem.
\( \nabla g_0(x) < 0 \) \hspace{1cm} (44a)
\( \nabla_i g_i(x) > 0 \quad \forall i \) \hspace{1cm} (44b)

and

\( \nabla_i g_j(x) \leq 0 \quad \forall i, j : j \neq i \) \hspace{1cm} (44c)
\( \nabla_j g_j(x) \geq 0 \quad \forall i, j : j \neq i \) \hspace{1cm} (44f)

or

\( \nabla_i g_0(x) = \nabla_j g_0(x) \quad \forall i, j \) \hspace{1cm} (44e)
\( \nabla_i g_j(x) > \sum_{j \neq i} |\nabla_i g_j(x)| \quad \forall i \) \hspace{1cm} (44g)

or

\[ g_0(x) < 0 \in \mathbb{R} \] \hspace{1cm} (44h)
\[ \frac{\Delta}{\tilde{\delta} + \Delta} \nabla_j g_j(x) > \sum_{j \neq i} |\nabla_i g_j(x)| \quad \forall j \] \hspace{1cm} (44i)

where \( \tilde{\delta} = \min_{i, x \in D} \nabla_i g_0(x) \)
and \( \Delta = \max_{i, x \in D} \nabla_i g_0(x) \)

The different cases above makes sure that the functions \( f_i \) of Eq. (41) fulfill cases very similar to cases (i-iii) of the qualifying conditions 4. Case (iv) however is new new to this thesis. Therefore, the conditions of Theorem 12 can be as follows.

**Theorem 13.** The following conditions guarantee problem (40) is F-L by case (iv) of the qualifying conditions.

\( \nabla g_0(x) < 0 \) \hspace{1cm} (45a)
\( \nabla_i g_0(x) = \nabla_j g_0(x) \quad \forall i, j \) \hspace{1cm} (45b)
\( \nabla_j g_j(x) > \sum_{i \neq j} |\nabla_i g_j(x)| \quad \forall i \) \hspace{1cm} (45c)
\( \nabla_i g_i(x) - \sum_{j \neq i} |\nabla_i g_j(x)| > \frac{1}{2} \max_{k, x \in D} \nabla_k g_k(x) \quad \forall i \) \hspace{1cm} (45d)

**Proof.** For case (iv) to hold, conditions (0.a), (0.b), (iv.a) and (iv.a) must hold simultaneously. It is straightforward to verify that conditions (45a) and (45b) ensures conditions (0.a) and (iv.a). To verify conditions (0.b)
and (iv.b), choose all \( \gamma_j \) equal:
\[
\gamma_j = \gamma = \frac{1}{\max_{k, x \in D} \nabla_k g_k(x)}
\]
(46)

Condition (45c) guarantees \( \gamma \) is finite and positive.

Now, as \( \nabla_i f_j(x) = \delta_{ij} - \gamma_j \nabla_i g_j(x) \),
\[
||\nabla f||_1 = \max_j \sum_{i \neq j} |\nabla_i f_j| < 1
\]
can be written as
\[
||\nabla f||_1 = \max_j \left\{ |1 - \gamma_j \nabla_j g_j| + \sum_{i \neq j} |\gamma_i \nabla_i g_j| \right\} < 1.
\]
(47)

The choice of \( \gamma_j \) in (47) ensures \( 1 - \gamma_j \nabla_j g_j \geq 0 \) for all \( j \), wherefore the absolute value sign can be removed. The equation above is therefore true if
\[
1 - \gamma_j \nabla_j g_j + \sum_{i \neq j} |\gamma_j \nabla_i g_j| < 1 \quad \forall j.
\]

With \( \gamma_j \) given by (47), this can be written
\[
\gamma \left( \nabla_j g_j + \sum_{i \neq j} |\nabla_i g_j| \right) > 0
\]
which, since \( \gamma > 0 \), is ensured by (45c).

Condition (iv.b) can be shown in a similar way. Again, let \( \gamma_j \) be given by eq. (45d). The condition requires that \( ||\nabla f||_\infty < 1/2 \), i.e.,
\[
||\nabla f||_\infty = \max_i \left\{ |1 - \gamma_i \nabla_i g_i| + \sum_{j \neq i} |\gamma_j \nabla_i g_j| \right\} < \frac{1}{2}.
\]
(48)

This is equivalent to
\[
|1 - \gamma_i \nabla_i g_i| + \sum_{j \neq i} |\gamma_j \nabla_i g_j| < \frac{1}{2} \quad \forall i.
\]

Inserting \( \gamma_i = \gamma_j = \gamma \) and removing the fist absolute value sign (by the same reasoning as above) gives
\[
-\gamma \nabla_i g_i + \sum_{j \neq i} |\gamma \nabla_i g_j| < -\frac{1}{2} \quad \forall i.
\]

Since both \( \gamma \) and \( \nabla_i g_i \) are positive, the equation above can be written as
\[
\nabla_i g_i - \sum_{j \neq i} |\nabla_i g_j| > \frac{1}{2 \gamma} = \frac{1}{2} \max_{k, x \in D} \nabla_k g_k(x) \quad \forall i.
\]

This is exactly condition (45d).
7.6 Fewer constraints than variables

The standard form of the F-Lipschitz problem (9) requires one (and only one) constraint for each variable. This section covers the case when the number of constraints are fewer than the number of variables. For simplicity we consider inequality constraints only. This is done without loss of generality, as a result of Proposition 8.

Consider a partitioned variable vector \([x^T \ z^T]^T \in D \subset \mathbb{R}^n\) and the problem

\[
\begin{aligned}
\max & \quad f_0(x, z) \\
\text{s.t.} & \quad x \leq f_x(x, z) \\
& \quad x^T \ z \in D
\end{aligned}
\tag{49}
\]

with

\[
D = \left\{ \begin{bmatrix} x \\ z \end{bmatrix} : \begin{bmatrix} a_x \\ a_z \end{bmatrix} \leq \begin{bmatrix} x \\ z \end{bmatrix} \leq \begin{bmatrix} b_x \\ b_z \end{bmatrix} \right\}.
\]

Even though problem (49) is not of the form required by the definition, an equivalent problem can be shown to be F-L.

**Proposition 14.** Consider problem (49). If

\[
\hat{f}_0(x) = f_0(x, b_z) \quad \text{and} \quad \hat{f}(x) = f_x(x, b_z)
\]

fulfill conditions 4 and 5, then the problem has a unique Pareto optimal point

\[
\begin{bmatrix} x^* \\ z^* \end{bmatrix} = \begin{bmatrix} f_x(x^*, b_z) \\ b_z \end{bmatrix}.
\]

**Proof.** By enforcing the second half of \(D\) twice, we get the equivalent problem

\[
\begin{aligned}
\max & \quad f_0(x, z) \\
\text{s.t.} & \quad x \leq f_x(x, z) \\
& \quad z \leq f_z(x, z) = b_z \\
& \quad x^T \ z \in D.
\end{aligned}
\tag{50}
\]

This problem has the right form, with \(f = \begin{bmatrix} f_x \\ f_z \end{bmatrix}\) and is F-L if

\[
\nabla f = \begin{bmatrix} \nabla_x f_x & \nabla_x f_z \\ \nabla_z f_x & \nabla_z f_z \end{bmatrix} \quad \text{and} \quad \nabla f_0 = \begin{bmatrix} \nabla_x f_0 \\ \nabla_z f_0 \end{bmatrix}
\]

fulfill conditions 4 and 5.
Since \( f_z = b_z \) is constant, the constraint gradient matrix simplifies to the special form
\[
\nabla f = \begin{bmatrix}
\nabla_x f_x & 0 \\
\nabla_z f_x & 0 
\end{bmatrix}.
\]

The two zero blocks are both non-negative and non-positive, wherefore the fulfill case (i) and (ii) of the qualifying conditions. Moreover, they add nothing to either of \( || \cdot ||_1 \) or \( || \cdot ||_\infty \) and need not be considered when verifying the qualifying conditions.

As a result of proposition 8, problem (50) is equivalent to
\[
\max \ f_0(x, z) \\
\text{s.t.} \ x \leq f_x(x, z) \\
z = f_z(x, z) = b_z.
\]

Note that the equality constraint above requires \( z = b_z \) wherefore \( D \) can be replaced with
\[
\hat{D} = \left\{ \begin{bmatrix} x \\ z \end{bmatrix} : \begin{bmatrix} a_x \\ b_z \end{bmatrix} \leq \begin{bmatrix} x \\ z \end{bmatrix} \leq \begin{bmatrix} b_x \\ b_z \end{bmatrix} \right\} = \left\{ \begin{bmatrix} x \\ z \end{bmatrix} : a_x \leq x \leq b_x, \ z = b_z \right\}
\]

without changing the feasible region. This is beneficial since the qualifying conditions involving \( f_x \) and \( f_0 \) only need to be fulfilled within \( D \), or in this case, when \( [x^T \ z^T]^T \in D \) and \( z = b_z \).

\[
7.6.1 \text{ Variations of qualifying conditions}
\]

When the constraint gradient matrix has the simplified form
\[
\nabla f = \begin{bmatrix}
\nabla_x f_x & 0 \\
\nabla_z f_x & 0 
\end{bmatrix}
\]

the qualifying and uniqueness conditions may be somewhat relaxed. In this subsection, two altered qualifying conditions are investigated. We start with a proposition, which we prove in an investigative way.

\textbf{Proposition 15.} Consider problem (49). If the subproblem consisting of
\( \nabla_x f_x \) and \( \nabla_x f_0 \) is F-L, then the entire problem is F-L if either
\[
\nabla_x f_0(x) \geq 0 \quad \text{and} \quad \nabla_x f_x(x) \text{ has no all-zero rows},
\]
or
\[
\text{all components of } \nabla_x f_0(x) \text{ are equal,} \\
\nabla_x f_0(x) \geq 0, \\
||\nabla_x f_x||_{\infty} < \frac{1}{2} \quad \text{and} \\
\nabla_x f_x(x) \geq 0.
\]

**Proof.** The argumentation is built on small modifications of the main theorem proof in section 7.2.1, which the reader is assumed to have read.

The particular form and partitioning of \( \nabla f \) will remain in \( A \) and \( c \):

\[
A = \begin{bmatrix}
\int_0^1 \nabla_x f_x(y + tr)dt & 0 \\
\int_0^1 \nabla_x f_x(y + tr)dt & 0
\end{bmatrix} = \begin{bmatrix}
A_{11} & 0 \\
A_{21} & 0
\end{bmatrix}
\]

and

\[
c = \begin{bmatrix}
\int_0^1 \nabla_x f_0(y + tr)dt \\
\int_0^1 \nabla_x f_0(y + tr)dt
\end{bmatrix} = \begin{bmatrix}
c_1 \\
c_2
\end{bmatrix}.
\]

Just like in the main proof, we need to show that \( \lambda = (I - A)^{-1} c \) is positive/non-negative. This time however, we will make use of the block structure of \( A \) and \( c \). Formulas for the inverse of a block matrix, as well as the products of two block matrices can be found in [4].

First we use \( \lambda = (I - A)^{-1} c \) (written in the form of case (i)). From the block matrix inverse formula:

\[
\lambda = (I - A)^{-1} c = \begin{bmatrix}
B_{11} & B_{12} \\
B_{21} & B_{22}
\end{bmatrix} \begin{bmatrix}
c_1 \\
c_2
\end{bmatrix},
\]

where \( B_{11} = (I - A_{11})^{-1} \), \( B_{12} = 0 \), \( B_{21} = A_{21}B_{11} \) and \( B_{22} = I \). Therefore,

\[
\lambda = \begin{bmatrix}
B_{11}c_1 \\
B_{12}c_1 + c_2
\end{bmatrix} = \begin{bmatrix}
B_{11}c_1 \\
A_{21}B_{11}c_1 + c_2
\end{bmatrix}.
\]

For a unique optimum, both components of \( \lambda \) must be positive/non-negative. Note that the first component is \( \lambda_1 = B_{11}c_1 \) or

\[
\lambda_1 = (I - A_{11})^{-1} c_1
\]
i.e. $\lambda_1$ is positive/non-negative if the sub-problem, consisting of $A_{11}$ and $c_1$, is F-L. This does not necessarily have to be through case (i), any way of showing that the subproblem is F-L (i.e. $\lambda_1 > 0$) will do.

The second component contains $\lambda_1$:

$$\lambda_2 = A_{21}B_{11}c_1 + c_2 = A_{21}\lambda_1 + c_2$$

Unique optimality requires $\lambda > 0$. For each part of $\lambda$ this means:

- $\lambda_1 > 0$: The sub-problem $(A_{11}, c_1)$ must be uniquely F-Lipschitz.
- $\lambda_2 > 0$: Since $\lambda_1 > 0$ it is sufficient that $A_{21} \geq 0$ and either:
  - $c_2 > 0$ (this is the current case (i), when the uniqueness condition holds), or
  - $c_2 \geq 0$ and every row of $A_{21}$ contain at least one non-zero component. As in the main proof, this is true if $\nabla_z f_0 \geq 0$ and $\nabla_z f_x$ contains no all-zero rows.

Next, as in case (ii) of the main theorem proof, we write

$$\lambda = (I - A^2)^{-1}(I + A) \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

Successive use of the partitioning (52) and the block matrix formulas gives

$$\lambda = \begin{bmatrix} C_{11}(I + A_{11}) \\ A_{21}A_{11}C_{11}(I + A_{11}) + A_{21}I \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} C_{11}(I + A_{11})c_1 \\ (A_{21}A_{11}C_{11}(I + A_{11}) + A_{21})c_1 + c_2 \end{bmatrix},$$

where

$$C_{11} = (I - A^2_{11})^{-1}.$$ 

This means

$$\lambda_1 = (I - A^2_{11})^{-1}(I + A_{11})c_1$$

and $\lambda_1$ is again positive if the subproblem, with $A_{11}$ and $c_1$ is F-L. The second component is more complicated,

$$\lambda_2 = \left(A_{21}A_{11}(I - A^2_{11})^{-1}(I + A_{11}) + A_{21}\right)c_1 + c_2 = A_{21}A_{11}\lambda_1 + A_{21}c_1 + c_2.$$
If the whole problem is F-L by case (ii), i.e. $A_{21} \leq 0$, $\|A_{21}\|_{\infty} < 1$ and the cs are parallel to the one vector, then

$$\lambda_2 = \frac{A_{21} A_{11}}{\leq 0, \leq 0, >0} \frac{\lambda_1 + c (A_{21} 1 + 1)}{>0} > 0$$

as expected. This however is already shown in the main proof, and we are interested in something not covered by case (ii). Let instead $A_{21} \geq 0$, $c_2 \geq 0$ and write

$$\lambda_2 = A_{21} \left( A_{11} (I - A_{11}^2)^{-1} (I + A_{11}) + I \right) c_1 + c_2. \quad (53)$$

This is (strictly) positive if $A_{21}$ has no all-zero rows and $\|A_{11} (I - A_{11}^2)^{-1} (I + A_{11}) + I\|_{\infty} > 0$.

Let $c_1 = \alpha 1$ be parallel to the one vector. Using the technique from case (ii) of the main theorem proof, the equation above is positive when $\|A_{11} (I - A_{11}^2)^{-1} (I + A_{11})\|_{\infty} < 1$.

Since $\|A_{11}\|_{\infty} < \|A\|_{\infty} < 1$, we can write the middle factor as a geometric series

$$\|A_{11} (I - A_{11}^2)^{-1} (I + A_{11})\|_{\infty} =$$

$$\|A_{11} \sum_{k=0}^{\infty} (A_{11}^2)^k (I + A_{11})\|_{\infty} =$$

$$\|A_{11} (I + A_{11}^2 + A_{11}^4 + \ldots) (I + A_{11})\|_{\infty} =$$

$$\|A_{11} ((I + A_{11}^2 + A_{11}^4 + \ldots) + (A_{11} + A_{11}^3 + A_{11}^5 + \ldots))\|_{\infty} =$$

$$\|A_{11} (I + A_{11} + A_{11}^2 + A_{11}^3 + \ldots)\|_{\infty} = \|A_{11} \sum_{k=0}^{\infty} A_{11}^k\|_{\infty}.$$

Using the basic norm inequalities of Chapter 3 and the geometric series (which converges as $\|A_{11}\|_{\infty} < 1$) we get

$$\|A_{11} \sum_{k=0}^{\infty} A_{11}^k\|_{\infty} \leq \|A_{11}\|_{\infty} \cdot \sum_{k=0}^{\infty} \|A_{11}\|_{\infty}^k = \frac{\|A_{11}\|_{\infty}}{1 - \|A_{11}\|_{\infty}}.$$

Finally,

$$\frac{\|A_{11}\|_{\infty}}{1 - \|A_{11}\|_{\infty}} < 1 \iff \|A_{11}\|_{\infty} < 1 - \|A_{11}\|_{\infty}$$

$$\iff \|A_{11}\|_{\infty} < \frac{1}{2}.$$

62
To summarize, if \( \lambda_1 > 0 \) (subproblem is F-L) and \( ||A_{11}||_\infty < 1/2 \), then
\[
||A_{11} (I - A_{11}^2)^{-1} (I + A_{11})||_\infty < 1,
\]
wherefore
\[
\left( A_{11} (I - A_{11}^2)^{-1} (I + A_{11}) + I \right) c_1 > 0.
\]
\[||A_{11}||_\infty < 1/2 \] if \( ||\nabla_x f_x||_\infty < 1/2 \). If it also holds that \( A_{21} \geq 0 \) and \( c_2 \geq 0 \) (which is true if \( \nabla_z f_x \geq 0 \) and \( \nabla_z f_0 \geq 0 \)), then \( \lambda_2 \) is also positive by equation (53). As
\[\lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix} > 0,\]
the entire problem is F-L.

7.7 Missing variables in cost function

Sometime it is practical or necessary to formulate problems where not all variables appear in the objective function. This results in problems not on the form (9). This section gives some sufficient conditions for these problems to have the properties of F-L.

Consider a partitioned variable and a problem on the form
\[
\begin{align*}
\text{max} & \quad f_0(x) \\
\text{s.t.} & \quad x \leq f_k(x, z) \\
& \quad z \leq f_\ell(x, z) \\
& \quad \begin{bmatrix} x \\ z \end{bmatrix} \in \mathcal{D} \subset \mathbb{R}^n.
\end{align*}
\tag{54}
\]
The variable \( z \) effects the objective only indirectly, through the first constraint. Redefining \( f_0 = f_0(x, z) \) gives a problem of the right form (9), but condition (5) will never hold since \( \nabla_x f_0(x, z) = 0 \) identically.

By partitioning also the constraint gradients and the cost function one can find circumstances when problem (54) behaves like a F-L problem, i.e. when it has a unique Pareto optimum at
\[
\begin{bmatrix} x^* \\ z^* \end{bmatrix} = \begin{bmatrix} f_k(x^*, z^*) \\ f_\ell(x^*, z^*) \end{bmatrix}.
\]

**Proposition 16.** Consider problem (54). If
\[
\nabla f(x, z) = \begin{bmatrix} \nabla_x f(x, z) \\ \nabla_z f(x, z) \end{bmatrix} \geq 0, \quad ||\nabla f||_1 < 1,
\]

63
\[ \nabla f_0(x) > 0 \quad \text{and} \quad \nabla_z f_x(x, z) \text{ has non-zero rows, then problem (54) has a unique Pareto optimum at} \]

\[
\begin{bmatrix}
x^* \\
z^*
\end{bmatrix} = \begin{bmatrix}
f_x(x^*, z^*) \\
f_z(x^*, z^*)
\end{bmatrix}.
\]

**Proof.** With the gradients of the (scalarized) cost function and constraint function partitioned as

\[ \nabla (\mu^T f_0) = \begin{bmatrix} \nabla_x f_0 \\
abla_z f_0 \end{bmatrix} \mu = \begin{bmatrix} \nabla_x f_0 \mu \\
0 \end{bmatrix} \]

and

\[ \nabla f = \begin{bmatrix} \nabla_x f_x \\
\nabla_x f_z \end{bmatrix}, \]
the variables \( A \) and \( c \) of the main theorem proof (of section 7.2.1) becomes

\[ A = \begin{bmatrix} A_{11} & A_{12} \\
A_{21} & A_{22} \end{bmatrix} \quad \text{and} \quad c = \begin{bmatrix} c_1 \\
0 \end{bmatrix}. \]

Case (i) of the proof requires \( \lambda = (I - A)^{-1} c > 0 \). Using the rules for block matrix inversion this gives

\[ (I - A)^{-1} = \begin{bmatrix} I - A_{11} & A_{12} \\
A_{21} & I - A_{22} \end{bmatrix}^{-1} = \begin{bmatrix} B_{11} & B_{12} \\
B_{21} & B_{22} \end{bmatrix}, \]

where

\[ B_{11} = \left( I - \left( A_{11} + A_{12} (I - A_{22})^{-1} A_{21} \right) \right)^{-1} \]

and

\[ B_{21} = (I - A_{22})^{-1} A_{21} B_{11}. \]

The values of \( B_{12} \) and \( B_{22} \) are of no interest since

\[ \lambda = \begin{bmatrix} B_{11} & B_{12} \\
B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} c_1 \\
0 \end{bmatrix} = \begin{bmatrix} B_{11} c_1 \\
B_{21} c_1 \end{bmatrix}. \]

As in the main proof, assuming \( ||\nabla f||_1 < 1 \) and \( \nabla f \geq 0 \) gives \( (I - A)^{-1} = B \geq 0 \). Therefore, since \( c_1 > 0 \), the first component of \( \lambda \) is positive if \( B_{11} \) have non-zero rows. This is always the case since \( B_{11} \) is an inverse and therefore has linearly independent rows (wherefore they are also non-zero).

The second component contains the first,

\[ \lambda_2 = (I - A_{22})^{-1} A_{21} B_{11} c_1. \]

When \( B_{11} c_1 = \lambda_1 > 0 \), \( A_{21} B_{11} c_1 > 0 \) if \( A_{21} \) is non-negative with non-zero rows. This gives \( \lambda_2 > 0 \) since \( (I - A_{22})^{-1} \geq 0 \) is an inverse wherefore it has non-zero rows.

\[ \square \]
Even for functions where all variables appear in the objective, this reasoning can be of use. The following corollary relaxes the uniqueness condition (5) when case (i) of the qualifying conditions hold.

**Corollary 17.** Proposition 16 still holds if $\nabla z f_0$ is not identically zero but only non-negative, i.e., $\nabla z f_0 \geq 0$.

**Proof.** In this case

$$\lambda = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} B_{11}c_1 + B_{21}c_2 \\ B_{21}c_1 + B_{22}c_2 \end{bmatrix}. $$

The positivity of $\lambda$ is still guaranteed by the first terms of each component, as in the proof of Proposition 16. The second terms are non-negative and can safely be ignored. \qed

### 7.8 Examples

**A first example**  Consider the problem

$$\begin{align*}
\max_x & \quad x_1 + 2x_2 \\
\text{s.t.} & \quad x_1 \leq 0.7 + ax_1 x_2 \\
& \quad x_2 \leq 0.3 (1 + x_1 + x_2) \\
& \quad x \in D = \{x : 0 \leq x \leq 1\},
\end{align*}$$

(55)

where $0 < a < 1/2$. The problem is not convex but has the correct F-L form (9),

$$\begin{align*}
\max_x & \quad f_0(x) \\
\text{s.t.} & \quad x \leq f(x) \\
& \quad x \in D = \{x : 0 \leq x \leq 1\}.
\end{align*}$$

In this case $f_0$ is scalar and will be denoted $f_0$. This gives

$$f_0(x) = x_1 + 2x_2$$

and

$$f(x) = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{bmatrix} = \begin{bmatrix} 0.7 + ax_1 x_2 \\ 0.3 (1 + x_1 + x_2) \end{bmatrix}. $$

In order to verify that problem (55) is F-L, we must check that $f$ and $f_0$ fulfill the qualifying and uniqueness conditions (conditions 4 and 5).

Collecting the partial derivatives of $f$ in the gradient gives

$$\nabla f(x) = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_1} \\ \frac{\partial f_1(x)}{\partial x_2} & \frac{\partial f_2(x)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} ax_2 & 0.3 \\ ax_1 & 0.3 \end{bmatrix}. $$
The gradient of \( f_0 \) is
\[
\nabla f_0(x) = \begin{bmatrix} 1 \\ 2 \end{bmatrix}
\]
and is positive for all \( x \). This means condition (0.a) is satisfied. To fulfill condition (0.b), we need \( ||\nabla f||_1 < 1 \). In this case,
\[
||\nabla f(x)||_1 = \max_j \sum_i |\frac{\partial f_j(x)}{\partial x_i}| = \max_j |\nabla f_j(x)| = \max \{ |ax_2| + |ax_1|, |0.3| + |0.3| \} = \max \{ a (|x_2| + |x_1|) , 0.6 \}.
\]
(56)
The condition must hold for all \( x \) in \( D \). Therefore it is required that
\[
\max \left\{ a \max_{x \in D} (|x_2| + |x_1|) , 0.6 \right\} < 1.
\]
This is true when \( a < 1/2 \).
In addition to (0.a) and (0.b), the problem must also fulfill one of the cases (i-iv). In this case, case (i) is fulfilled since \( \nabla f(x) \geq 0 \) for all \( x \in D \). It is now verified that problem (55) fulfills the qualifying conditions. Theorem 7 therefore guarantees that \( x^* = f(x^*) \) is an optimal solution. Since \( \nabla f_0(x) > 0 \) holds for all \( x \) it certainly holds for \( x^* \). The uniqueness condition is therefore fulfilled and \( x^* \) is guaranteed to be the unique optimum.
In this case, equation (57) has an analytic solution:
\[
\begin{align*}
x_1^* &= \frac{7 - 3a - \sqrt{9a^2 - 84a + 49}}{6a} \\
x_2^* &= \frac{3 + 7 - \sqrt{9a^2 - 84a + 49}}{14a}.
\end{align*}
\]
As with every F-L problem, the solution could also be obtained through repeated iterations of the constraint function,
\[
x^{k+1} = f(x^k).
\]
(58)
The initial point \( x^0 \) can be any point in \( D \), not necessarily feasible.
Figure 1: The feasible region is shown with dashed lines. The solid line marks the sequence $x^k$.

Figure 1 shows the feasible region and the sequence $x^k$ generated by (58) when $a = 0.3$. The initial point was chosen in the center of $\mathcal{D}$,

$$x^0 = \begin{bmatrix} 0.5 \\ 0.5 \end{bmatrix}.$$ 

As discussed in section 7.4, $||x^k - x^*||_\infty$ is guaranteed to decrease faster than $\alpha^k ||x^0 - x^*||_\infty$, where

$$\alpha = \max_{x \in \mathcal{D}} ||\nabla f(x)||_1 = 2a = 0.6.$$ 

This worst-case bound on convergence, and $\varepsilon_k = ||x^k - x^*||_\infty$ is shown in Figure 2. It can be seen that the sequence converges to an accuracy of $10^{-4}$ in 9 iterations.
Example 2  This example considers the problem

$$\max_x f_0(x) = e^{x_1 + x_2}$$

s.t.  \( x_1 \leq 1 - 0.5 \sin(x_1) - 0.4x_2 \)

\( x_2 \leq 1 - 0.2x_1 \)

\( 0 \leq x \leq 1. \) (59)

Condition (0.a) is fulfilled since

$$\nabla f_0(x) = e^{x_1 + x_2} \begin{bmatrix} 1 \\ 1 \end{bmatrix} > 0$$

is positive for all \( x \). Furthermore, we have

$$\nabla f(x) = \begin{bmatrix} -0.5 \cos(x_1) & -0.2 \\ -0.4 & 0 \end{bmatrix}$$

and

$$\|\nabla f(x)\|_1 = \max_j \sum_i |\nabla_i f_j(x)|$$

$$= \max \{ |-0.5 \cos(x_1)| + | -0.4|, |-0.2| \} \quad \text{when} \ x_1 = 0,$$

$$= 0.9 < 1,$$
wherefore condition \((0.b)\) is verified. This time the problem is not F-L by case \((i)\), since \(\nabla f \geq 0\).

Instead, \(\nabla f \leq 0\) and \(\nabla f_0(x) = \nabla^2 f_0(x) = e^{x_1+x_2}\) for all \(x\) in \(\mathcal{D}\), so conditions \((\text{ii.a})\) and \((\text{ii.b})\) hold. Condition \((\text{ii.c})\) is also satisfied since

\[
\|\nabla f(x)\|_\infty = \max_i \sum_j |\nabla_i f_j(x)| = \max \{|-0.5 \cos(x_1)| + |-0.2|, |-0.4|\}
\]

\[
= 0.7 < 1.
\]

Therefore, Problem \((59)\) is F-L by case \((\text{ii})\). An optimal point \(x^*\) is given by the solution to the system of equations

\[
x_1^* = 1 - 0.5 \sin(x_1^*) - 0.4 x_2^*
\]

\[
x_2^* = 1 - 0.2 x_1^*.
\]

Furthermore, as \(\nabla f_0 > 0\) for all \(x\), the uniqueness condition \((5)\) is fulfilled and \(x^*\) is also the unique optimal point. As in the previous example, \(x^*\) can be found as the limit of the sequence \(x^{k+1} = f(x^k)\).

An initial point \(x^0 = [0.5 \ 0.5]^T\) in the center of \(\mathcal{D}\) guarantees that the initial error \(\|x^0 - x^*\| \leq 1/2\). As \(\alpha = \|\nabla f(x)\|_\infty = 0.7\), the theory of Section 7.4 says we can find a point \(x^\bar{k}\) with a component-wise error less than, e.g., \(10^{-4}\) after at most

\[
\bar{k} = \left\lceil \frac{\ln 10^{-4} - \ln 0.5}{\ln 0.7} \right\rceil = 24
\]

iterations. Figure shows the feasible region and the sequence \(\{x^0, x^1, \ldots x^{24}\}\) converging to the optimal point \(x^*\).
Figure 3: The feasible region of Problem (59), and the sequence $x^k$ converging to the optimal point $x^*$. 
8 Convergence Speed Analysis

For most iterative methods, there exist some analysis guaranteeing convergence under certain circumstances. This corresponds to examining worst case scenarios, for example by bounding some norm or spectral radius (which can be seen as the asymptotic convergence speed) below one, making the iterations contractive. Other methods include Lyapunov-like arguments [8]. Comparing two different methods however, is not easily done.

In order to say that method A is faster than method B in general, one must show that the worst case of A is faster than the best case of B. This can be extremely conservative if one does not restrict the set of problems the comparison is performed over. As we want to show faster convergence of the F-L method, we must find an upper bound of the convergence speed of the other method. Such bounds are hard to find in literature, as the purpose of convergence analysis usually is to show convergence (i.e. lower bound the convergence speed).

An additional complication is that once a problem is shown to be F-L, solving it is no longer an optimization problem in the sense of finding an optimal point in the feasible set. For example, in contrast to almost every method of optimization, the properties of the objective function (once shown to be F-L) have no effect on the solution.

In the rest of this section we first solve a F-L problem by a simple traditional method, where the assumptions give an upper bound on convergence. We then solve it as a F-L and compare the convergence speeds, giving a criterion for faster convergence of the F-L method.

8.1 Problem formulation

We will start with a scalar problem on F-L form. In order to simplify the analysis, the problem will have only equality constraints:

\[
\max \quad -f_0(x) \\
\text{s.t.} \quad x = f(x) \in \mathbb{R}^n
\]

On canonical form the problem becomes

\[
\min \quad f_0(x) \\
\text{s.t.} \quad h(x) = x - f(x) = 0
\] (60)
The case of only equality constraints is somewhat unfortunate. In order for problem \((60)\) to be F-L, the \(f\) must be contractive and the fixed point \(x^* = f(x^*)\) will therefore be unique. This means that the problem only has one feasible point \((x^*)\). The optimization problem, i.e., picking the best element from the feasible set, therefore becomes trivial; there is only one point to choose from. The point must fulfill
\[
h(x) = 0, \tag{61}\]
wherefore the optimization problem collapses to solving a system of equations. This (reducing the number of choices to one) is precisely what usually gives F-L optimization an advantage.

However, it might not be apparent that equation \((61)\) has a unique solution. This is especially true if \(h\) does not have \(n\) components, as in section 7.6 (we will still consider \(n\) components, for simplicity).

### 8.2 First order Lagrangian method

The Lagrangian of the problem is given by
\[
L(x, \lambda) = f_0(x) + \lambda^T h(x)
\]
where
\[
\lambda \in \mathbb{R}^n, \quad \lambda \neq 0.
\]
Locally optimal \((x^*, \lambda^*)\) must fulfill
\[
\begin{cases}
\nabla_x L(x^*, \lambda^*) = \nabla f_0(x^*) + \nabla h(x^*) \lambda^* = 0 \\
h(x^*) = 0
\end{cases}. \tag{62}
\]
This is a system with \(2n\) equations in \(2n\) variables, whose solutions are the stationary points of the Lagrangian. The gradient method solves this by taking steps (of step length \(\alpha\)) in the direction of the negative gradient,
\[
x^{k+1} = x^k - \alpha \nabla_x L(x^k, \lambda^k) \tag{63a} \\
\lambda^{k+1} = \lambda^k + \alpha h(x^k). \tag{63b}
\]
For a shorter notation, introduce \(y = [x^T, \lambda^T]^T\) and
\[
M_\alpha(y) = \begin{bmatrix}
x - \alpha \nabla_x L(x, \lambda) \\
\lambda + \alpha \nabla_\lambda L(x, \lambda)
\end{bmatrix} = \begin{bmatrix}
x \\
\lambda
\end{bmatrix} - \alpha \begin{bmatrix}
\nabla f_0(x) + \nabla h(x) \lambda \\
-h(x)
\end{bmatrix}. \tag{64}
\]
for some small enough \(\alpha > 0\). Equations \((63)\) can now be written as
\[
y^{k+1} = M_\alpha(y^k) \tag{65}
\]
Clearly \(M_\alpha(y^*) = y^*\), where \(y^* = (x^*, \lambda^*)\). If \(M_\alpha(y)\) is contractive in some neighborhood around \(y^*\), any sequence \(y^{k+1} = M_\alpha(y^k)\) within this neighborhood will converge to \(y^*\).
8.2.1 Conditions for convergence

The following proposition is based on [8, (prop. 4.4.1)].

**Proposition 18.** Assume that \( y^* = M_\alpha(y^*) \) and that the eigenvalues of \( \nabla M_\alpha(y^*) \) have magnitudes strictly less than one. Then, for some \( y^0 \) sufficiently close to \( y^* \), the sequence \( y^k \) given by \( y^{k+1} = M_\alpha(y^k) \) will converge to \( y^* \).

**Proof.** Applying the mean value theorem to the \( i \)th component of \( M_\alpha \) gives

\[
[M_\alpha(y) - M_\alpha(y^*)]_i = \nabla M_{\alpha,i}(\hat{y}_i)^T (y - y^*)
\]

Here \( \nabla M_{\alpha,i}(\hat{y}_i)^T \) is the transpose of the gradient of the \( i \)th component of \( M_\alpha(\hat{y}_i) \) and \( \hat{y}_i \) lies somewhere on the line between \( y \) and \( y^* \). By collecting all \( \nabla M_{\alpha,i}(\hat{y}_i) \) as columns of the matrix \( R^T \), we can write

\[
M_\alpha(y) - M_\alpha(y^*) = R(y - y^*) \tag{65}
\]

With \( y \) close enough to \( y^* \) (and therefore \( \hat{y}_i \) close enough to \( y^* \)), \( R^T \) will be close enough to \( \nabla M_\alpha(y^*) \) and all eigenvalues of \( R \) will lie strictly within the unit circle, i.e.

\[ \rho(R) < 1. \]

This follows since \( R \) and \( R^T \) have the same eigenvalues, and the eigenvalues are continuous functions of \( R^T \).

Then, by proposition A.15 in [8], there exists a norm \( ||\cdot|| \) and a sphere \( S = \{ y : ||y - y^*|| < L \} \) with respect to this norm, such that \( ||R|| < 1 - \epsilon \) for some \( \epsilon > 0 \) when \( y \in S \).

If \( y^k \in S \), then

\[
||y^{k+1} - y^*|| = ||M_\alpha(y^k) - M_\alpha(y^*)|| \leq ||R|| ||y^k - y^*|| < ||y^k - y^*|| < L
\]

wherefore \( y^{k+1} \in S \). Therefore, if \( y^0 \in S \), all \( y^k \in S \) for all \( k \geq 0 \) and

\[
||y^{k+1} - y^*|| = ||M_\alpha(y^k) - M_\alpha(y^*)||
\]

\[
\leq ||R|| ||y^k - y^*|| = ||R|| \left(||R|| ||y^{k-1} - y^*||\right)
\]

\[
= ||R||^{k+1} ||y^0 - y^*||.
\]

This gives (since \( ||R|| < 1 \))

\[
\lim_{k \to \infty} ||y^{k+1} - y^*|| = \lim_{k \to \infty} ||R||^{k+1} ||y^0 - y^*|| = 0
\]

and \( y^{k+1} \) converges to \( y^* \) as \( k \to \infty \). \( \square \)
The proposition states that the method converges when $\rho(\nabla M_\alpha(y^*)) < 1$, and from the proof it is also clear that convergence becomes faster as the spectral radius becomes smaller. In fact, the spectral radius is sometimes known as the asymptotic convergence speed. Thus, in order to lower bound the convergence speed of this gradient method, we need the minimal spectral radius of $\nabla M_\alpha(y^*)$.

Taking the gradient of equation (64) gives

$$\nabla M_\alpha(y^*) = I - \alpha B,$$

where

$$B = \begin{bmatrix} \nabla_{xx} L(x^*, \lambda^*) & \nabla h(x^*) \\ -\nabla h(x^*)^T & 0 \end{bmatrix}$$

is the gradient of the update direction. Note that $B$ is a $m \times m$ matrix, where $m = 2n$.

The eigenvalues of $\nabla M_\alpha$ are $1 - \alpha \mu_i$ where $\mu_1 \geq \mu_2 \geq \cdots \geq \mu_m > 0$ are the eigenvalues of $B$. Therefore,

$$\rho(\nabla M_\alpha) = \rho_M(\alpha) = \max_i \{|1 - \alpha \mu_i|\} = \max_i \rho_M^i(\alpha).$$

In general, $\mu_i$ are complex and $\rho_M(\alpha)$ becomes the maximum of a collection of functions of the form $\sqrt{a_i \alpha^2 + b_i \alpha + 1}$. In this case it is difficult to say anything about $\min_\alpha \rho_M(\alpha)$ (remember, we want the minimal spectral radius) without knowing all the eigenvalues $\mu_i$ of $B$. We want a more practical condition and will therefore assume all eigenvalues are real.

Assuming the eigenvalues of $B$ are real, $\rho_M(\alpha)$ is the maximum of $i$ piecewise linear functions $\rho_M^i(\alpha)$. For $\rho(\nabla M_\alpha) < 1$, $B$ must be positive definite (i.e. $\mu_i > 0 \ \forall i$). In this case, finding the minimal spectral radius is a lot easier.
Figure 4: Three different $\pm \rho^i_M(\alpha)$ are shown. The slopes are the eigenvalues $\mu_i$. The first and last ($i = m$ and $i = 1$) correspond to the lowest and highest eigenvalue $\mu_i$. The middle represents any $\rho^i_M(\alpha)$ where $1 < i < m$.

The behavior of $\rho_M(\alpha)$ can be seen in Figure 4. When $\alpha \to 0$ all $\rho^1_M(\alpha) \to 1$ wherefore $\rho_M(\alpha) \to 1$. As $\alpha$ increases all $\rho^i_M(\alpha)$ will decrease linearly and as $-\mu_m$ is the least negative slope, $\rho_M(\alpha) = \rho^m_M(\alpha)$. All $\rho^1_M(\alpha)$ will continue to decrease until $1 - \alpha \mu_1 = 0$ ($\alpha = 1/\mu_1$) and $\rho^1_M(\alpha)$ starts to increase with slope $\mu_1$. After this point, no other $\rho^i_M(\alpha)$ will ever have a larger slope than $\rho^1_M(\alpha)$. The minimal $\rho_M(\alpha)$ occurs at the intersection of $\rho^1_M$ and $\rho^m_M$. Hence, the best possible $\hat{\alpha}$ is at the

$$\hat{\alpha} : 1 - \hat{\alpha} \mu_m = \hat{\alpha} \mu_1 - 1$$

$$\hat{\alpha} = \frac{2}{\mu_1 + \mu_m}$$

and

$$\rho_M(\hat{\alpha}) = \rho^m_M(\hat{\alpha}) = 1 - \mu_m \frac{2}{\mu_1 + \mu_m} = \frac{\mu_1 - \mu_m}{\mu_1 + \mu_m}$$

Let $\gamma_B = \mu_1/\mu_m$ be the condition number of $B$. Then, the best possible spectral radius of $\nabla M_\alpha$ is

$$\hat{\rho}_M = \rho_M(\hat{\alpha}) = \frac{\gamma_B - 1}{\gamma_B + 1}$$
\( \hat{\rho}_M \) is an increasing function of the condition number \( \gamma_B \). When \( \gamma_B = 1 \), \( \hat{\rho}_M = 0 \) and the method converges in one iteration. As \( \gamma_B \) grows towards infinity, \( \hat{\rho}_M \) will approach 1 and the convergence will stall.

**Eigenvalues of \( B \)**

As \( \gamma_B = \mu_1 / \mu_m \) depends on the eigenvalues of \( B \), these must be found. In general, the eigenvalues \( \nu_i \) of any matrix \( M \) appear in the identities

\[
\sum_i \nu_i = \text{trace}(M) \quad \text{and} \quad \prod_i \nu_i = \det(M). \tag{66}
\]

In this case

\[
M = B = \begin{bmatrix} Q & A \\ -A^T & 0 \end{bmatrix},
\]

where \( Q = \nabla_{xx}L(x^*, \lambda^*) \) and \( A = \nabla_h(x^*) \). The special structure of \( B \) gives

\[
\text{trace}(B) = \text{trace}(Q)
\]

and (using expressions for the determinant of a block matrix, found in [4])

\[
\det(B) = \det\left(\begin{bmatrix} Q & A \\ -A^T & 0 \end{bmatrix}\right) = \det(Q) \det(A^T Q^{-1} A)
= \det(Q) \det(A^T) \det(Q^{-1}) \det(A) = \det(QQ^{-1}) \det(A^T A)
= \det(A^T A) = \det(A)^2 = \det(A^2).
\]

The identities in (66) can be used to derive bounds on the eigenvalues. We have

\[
\sum_{i=1}^m \mu_i = \text{trace}(B) = \text{trace}(Q)
\]

wherefore

\[
\text{trace}(Q) = \sum_{i=1}^m \mu_i \begin{cases} \leq m \mu_1 & \Leftrightarrow \mu_1 \geq \frac{\text{trace}(Q)}{m} \\ \geq m \mu_m & \Leftrightarrow \mu_m \leq \frac{\text{trace}(Q)}{m} \end{cases}.
\]

In the same manner we have \( \prod_i \lambda_i = \det(B) = \det(A^2) \) and

\[
\det(A^2) = \prod_{i=1}^m \mu_i \begin{cases} \leq (\mu_1)^m & \Leftrightarrow \mu_1 \geq \sqrt[m]{\det(A^2)} \\ \geq (\mu_m)^m & \Leftrightarrow \mu_m \leq \sqrt[m]{\det(A^2)} \end{cases}.
\]

Note that the earlier assumption \( \mu_i > 0 \) gives \( \sqrt[m]{\det(A^2)} > 0 \) and \( \text{trace}(Q) > 0 \).

We can now bound the condition number of \( B \) in two ways:

\[
\gamma_B = \frac{\mu_1}{\mu_m} \geq \frac{\sqrt[m]{\det(A^2)}}{\text{trace}(Q)} = \frac{m \sqrt[m]{\det(A^2)}}{m \text{trace}(Q)} = \frac{\sqrt[m]{\det(A^2)}}{\text{trace}(Q)} = b
\]

76
and
\[ \frac{\mu_1}{\mu_m} \geq \frac{\text{trace}(Q)}{\sqrt{\det(A^2)}} = \frac{\text{trace}(Q)}{m \sqrt{\det(A^2)}} = \frac{1}{b}. \]

There is also the trivial bound \( \gamma_B \geq 1 \), that holds for any condition number by definition. However, out of \( b \) and \( b^{-1} \), at least one will be greater or equal to one. This means
\[ \gamma_B \geq \max \left\{ b, \frac{1}{b} \right\} = \max \left\{ \frac{m \sqrt{\det(A^2)}}{\text{trace}(Q)}, \frac{\text{trace}(Q)}{m \sqrt{\det(A^2)}} \right\} \quad (67) \]

and the a bound on the best possible spectral radius of \( \nabla M_\alpha \) is
\[ \hat{\rho}_M = \gamma_B - 1 \frac{1}{\gamma_B + 1} \]
\[ \geq \max \left\{ \frac{m \sqrt{\det(A^2)} - \text{trace}(Q)}{m \sqrt{\det(A^2)} + \text{trace}(Q)}, \frac{\text{trace}(Q) - m \sqrt{\det(A^2)}}{\text{trace}(Q) + m \sqrt{\det(A^2)}} \right\} \]
\[ = \frac{\sqrt{\det(A^2)} - \text{trace}(Q)}{\sqrt{\det(A^2)} + \text{trace}(Q)}. \]

To summarize, the convergence speed of this method depends on the spectral radius \( \rho_M(\alpha) = \rho(\nabla M_\alpha(y^*)) \). This in turn depends on the eigenvalues of \( B \) and the parameter \( \alpha \). If the optimal \( \alpha \) is chosen in each iteration, the best possible spectral radius
\[ \hat{\rho}_M = \frac{\gamma_B - 1}{\gamma_B + 1}, \]
will still be higher than
\[ \frac{\sqrt{\det(I - \nabla f(x^*))}^2 - \text{trace}(\nabla^2 f_0(x^*) + \sum_k \nabla^2 f_k(x^*) \lambda_k^*)}{\sqrt{\det(I - \nabla f(x^*))}^2 + \text{trace}(\nabla^2 f_0(x^*) + \sum_k \nabla^2 f_k(x^*) \lambda_k^*)}. \]

8.3 Comparison With F-L Method

If we know the problem is F-L, we know the solution fulfills \( x^* = f(x^*) \) and that \( f \) is contractive. There are several ways of solving this system of equations (which is half the size of Eq. (62)). By the argument of the previous subsection, we are interested in the asymptotic convergence speed \( \rho_f = \rho(\nabla f(x^*)) \). In this sense, the F-L method is faster than the Lagrangian method if \( \rho_f \) is smaller than the smallest possible \( \hat{\rho}_M \), that is if
\[
\frac{|m \sqrt{\det(A^2)} - \text{trace}(Q)|}{m \sqrt{\det(A^2)} + \text{trace}(Q)} = \\
\frac{|m \sqrt{\det(I - \nabla f(x^*))^2 - \text{trace}((\nabla^2 f_0(x^*) + \sum_k \nabla^2 f_k(x^*)\lambda_k^*)|}{m \sqrt{\det(I - \nabla f(x^*))^2 + \text{trace}((\nabla^2 f_0(x^*) + \sum_k \nabla^2 f_k(x^*)\lambda_k^*)|} \geq \rho_f. \quad (68)
\]

Assume \(m \sqrt{\det(A^2)} \geq \text{trace}(Q)\). Then, the absolute value can be removed and (68) simplifies to

\[
m \sqrt{\det(A^2)} - \text{trace}(Q) \geq \rho_f \cdot (m \sqrt{\det(A^2)} + \text{trace}(Q)).
\]

or

\[
m \sqrt{\det(A^2)} \frac{1 - \rho_f}{1 + \rho_f} \geq \text{trace}(Q). \quad (69)
\]

Note that the assumption holds if (69) holds. Now,

\[
\det(A^2) = \det(A^2) = \det(I - \nabla f(x^*))^2 = \prod_{i=1}^n (1 - \nu_i(\nabla f(x^*)))^2
\]

(where \(\nu_i\) are the eigenvalues of \(\nabla f(x^*)\)), so we can bound \(\sqrt{\det(A^2)}\) as

\[
(1 - \rho_f)^2 \leq m \prod_{i=1}^n (1 - \nu_i(\nabla f(x^*)))^2 \leq (1 + \rho_f)^2.
\]

This means (69), and therefore (68), is true if

\[
m (1 - \rho_f)^2 \frac{1 - \rho_f}{1 + \rho_f} \geq \text{trace}(Q). \quad (70)
\]

Now, assume instead that \(m \sqrt{\det(A^2)} < \text{trace}(Q)\). Analogous reasoning gives

\[
m \sqrt{\det(A^2)} \frac{1 + \rho_f}{1 - \rho_f} \leq \text{trace}(Q) \quad (71)
\]

(again, the new assumption holds if (71) holds) and

\[
\text{trace}(Q) \geq m (1 + \rho_f)^2 \frac{1 + \rho_f}{1 - \rho_f}. \quad (72)
\]

By combining (70) and (72), we have that (68) holds (guaranteeing F-L is faster) if

\[
\frac{\text{trace}(Q)}{m} \leq \frac{(1 - \rho_f)^3}{1 + \rho_f} \quad \text{or} \quad \frac{(1 + \rho_f)^3}{1 - \rho_f} \leq \frac{\text{trace}(Q)}{m}.
\]
That is, the F-L method is faster if

\[
\frac{\text{trace}(Q)}{2n} = \frac{1}{2n} \text{trace} \left( \nabla^2 f_0(x^*) + \sum_k \nabla^2 f_k(x^*) \lambda_k^* \right)
\]
does not lie in the interval

\[
\left( \frac{(1 - \rho_f)^3}{1 + \rho_f}, \frac{(1 + \rho_f)^3}{1 - \rho_f} \right).
\]

Not that for the extreme worst-case of the F-L method, when \( \rho_f \to 1 \), this interval becomes \((0, \infty)\). This makes sense, as we cannot guarantee faster F-L convergence if the F-L problem cannot be solved.

Both \( Q \) and \( \rho_f = \rho(\nabla f(x^*)) \) depend on the optimal points \( x^* \) and \( \lambda^* \). If the this criterion is to be used in practice, it is only interesting before the problem is actually solved (so \( x^* \) and \( \lambda^* \) will not be known). The spectral radius dependence of \( x^* \) can be circumvented by instead considering \( \max_{x \in D} \rho(\nabla f(x)) \). Avoiding the dual optimum \( \lambda^* \) is problematic, since it this far only is restricted by \( \lambda^* \neq 0 \). Perhaps a bound could be found through the KKT-conditions and known or assumed properties of the problem. As a last resort, one could consider the linearly constrained problem, where \( \nabla^2 f_k = 0 \) for all \( k \).
9 Conclusions and Future Work

9.1 Conclusions

In this paper the F-L theory has been extended in two directions.

The first direction expands the class of F-L problems. Contributions include the notion of weak F-L problems and case (iv) of the qualifying conditions, Proposition 8 and the Lagrangian proof as well as the relaxations of Section 7.2.2. The results in sections 7.6 and 7.8 are also new.

The second direction presents novel results comparing convergence speeds of a F-L approach and a traditional Lagrangian method. This result might not be directly applicable to real problems, but provides a first step for future research.

9.2 Future Work

F-L Theory Throughout Chapter 7 it is assumed that $f$ maps $D$ onto itself and that the components $f_i(x)$ are smooth for all $x$ in $D$. These assumptions are not always necessary and should be further investigated. Investigation on the non-smooth case is also required if one is to consider the case with more constraints than variables.

It is my belief that substantial extensions of the qualifying conditions remain to be found. One such possibility might be to use the Lagrangian proof in Section 7.2.1, where $A$ and $c$ are to be evaluated only at candidates for optimality (as opposed to all lines between feasible points in the mean value proof). It might be possible to specify smaller regions than $D$ where the conditions need to hold.

Furthermore, cases (ii) and (iv) of the qualifying conditions require $\nabla f_0$ parallel to $1$. It would be interesting to investigate if deviation from this (very restrictive) condition could be compensated by altering (toughening) the remaining conditions.

A generalization of all the cases might be to consider when $\nabla f_0$ belongs to other cones than $\mathbb{R}^m_+$. The number of problems treatable with F-L optimization can also increase by studying transforms to equivalent problems, as done in Section 7.2.2.
Convergence analysis The comparison of convergence speeds in this thesis is done with a simple first order method, under additional assumptions of real eigenvalues. I believe that it is possible to generalize this approach by considering a modified problem or by circumventing the need for real eigenvalues.

It would also be nice to be able to compare convergence with more sophisticated and widely used methods. Considering also inequality constrained problems might give stronger results, since they usually make the problem harder for traditional methods, while leaving the F-L approach unaffected.
References


