A trust region based nonlinear gradient projection method for constrained optimization in a model for the storage location assignment problem

Masterarbeit

zur Erlangung des akademischen Grades
Master of Science (MSc)

eingereicht von

Daniel Tschernutter, BSc
Matrikelnummer: 1110396

Betreuer(in): Ao.Univ.-Prof. Mag. Dr.rer.nat. Stephen Keeling

Institut für Mathematik und Wissenschaftliches Rechnen
der Karl Franzens Universität Graz

Juli 2017

Eidesstattliche Erklärung

Hiermit bestätige ich, dass die vorliegende Arbeit von mir selbständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen nicht benutzt und die den Quellen wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht wurden. Ebenso wurden die in der Arbeit dargestellten Daten von mir selbst erfasst sowie ausgewertet. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen inländischen oder ausländischen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht. Die vorliegende Fassung entspricht der eingereichten elektronischen Version.

Graz, im Juli 2017 Daniel Tschernutter, BSc

Abstract

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Kurzzusammenfassung

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Contents

1 Introduction

A warehouse is a facility in which inventories are stored. Generally speaking, there are two different types of warehouses, production warehouses and distribution centers. In distribution centers the products arrive by truck, rail or other modes of transportation, are unloaded and stocked until they are retrieved from their storage locations and transported to an order assembly area (see Ghiani et al. 2005, p. 157-198). The storage locations can be assigned in several ways. Two frequently used types of storage location assignment are the random storage policy and the dedicated storage policy. In case of random storage, every incoming good is assigned to one of the empty, or at least not full, storage locations with equal probability. This assignment policy results in high space utilization but typically increases travel distance. In the dedicated storage policy, each storage location is reserved for a single product, even if this product is currently out of stock. In combination with the so called full-turnover storage policy, which distributes products within the storage area according to their turnover, i.e. products with higher sales rates are located closer to the depot than those with lower, this policy results in a lower space utilization but, due to several reasons, has the advantage of a decrease in picking time (see Koster et al. 2006). The costs commonly involved in warehousing can be divided into receiving, holding, shipping and handling costs.

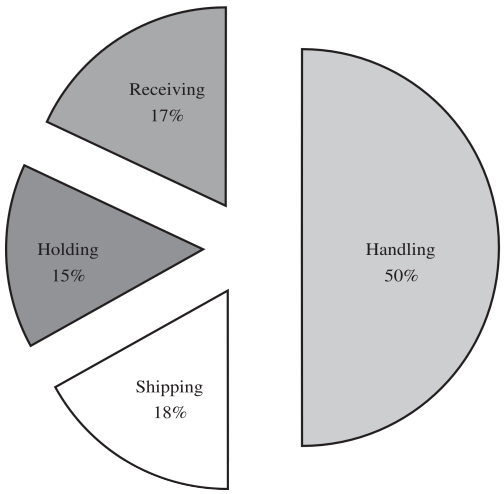


Figure 1: Common Warehouse Costs; Percentage of Receiving, Holding, Shipping and Handling Costs (Ghiani et al. 2005 [p.159])

As can be seen in figure 1, a special emphasis has to be put on the handling. A lot of research focuses on the optimization of the picking time by optimizing picking routes

through the use of several heuristics or optimal procedures (see e.g Petersen 1997). Another approach aims at assigning incoming goods to storage locations or storage zones in such a way that costs are reduced right from the start (see Koster et al. 2006). This thesis focuses on the aforementioned so-called storage location assignment problem. Its aim is to combine the advantages of both policies mentioned above, by finding a probability distribution for each product, which gives products with a high sales rate a higher chance to be assigned to a storage location near the depot than those with a lower sales rate. Additionally two more factors should be incorporated in those probability distributions. The first one is the product affinity or correlation. Products that are often commissioned together are called affine or correlated and it might be useful to store these products close to each other (see Kofler et al. 2011; Bindi et al. 2007). The second factor is the dispersion of the products inside a warehouse. If there are, for example, 50 units of an out of stock product to be stored inside a warehouse using the random storage policy it might happen that each of these 50 units get assigned to different storage locations, which are spread all over the warehouse. If now an order arrives which requests 5 units of this product, the picker has to visit 5 different storage locations to collect them, instead of, for example, only one. Of course there are picking policies, like, for example, batch picking (see Gu et al. 2007), where several orders are partitioned into batches, get then picked by different workers and are sorted afterwards, where such situations might not have an considerable impact on the picking time, but in a discrete order picking system, which is quite common because of its simplicity and reliability, where a single worker picks all items of a single order (see Eisenstein 2008), such situations can increase the picking time. Therefore, a probability distribution used to assign a product to a storage location should rather cluster the products inside the warehouse, i.e. assign a product more likely to neighbouring storage locations rather then spreading them all over the warehouse. Thus, the aim of this thesis is to find a probability distribution, defined on the storage locations of a given warehouse for each product to be stored in the warehouse, which considers the factors:

- product correlation
- product turnover
- product clustering

2 Modelling

2.1 Concepts of information theory

For the modelling of the problem, two concepts of information theory are needed. The first is the concept of entropy, or, more specifically, Shannon entropy, which is a measure of uncertainty. The second is the Kullback-Leibler divergence or relative entropy, which is a measure of the distance between two probability distributions. The definitions, theorems and proofs follow the ones in Cover et Thomas 2006.

Definition 1 (Entropy)**.** *Let X be a discrete random variable with possible values in $\mathfrak{X} = \{x_1, \ldots, x_n\}$ and let p be a probability mass function with $p(x) = \mathbb{P}(X = x)$ for $x \in \mathfrak{X}$. The entropy $H(X)$ of the discrete random variable X is defined as*

$$H(X) = -\sum_{x \in \mathfrak{X}} p(x) \log_2(p(x))$$

with the convention $0 \ \log_2(0) = 0$.

Since the entropy only depends on the probability mass function p, the notation $H(p)$ is also common for the quantity above.

Definition 2 (Kullback-Leibler divergence or relative entropy)**.** *Let p and q be two probability mass functions defined on $\mathfrak{X} = \{x_1, \ldots, x_n\}$. The Kullback-Leibler divergence or relative entropy between p and q is defined as*

$$D(p||q) = \sum_{x \in \mathfrak{X}} p(x) \log_2 \left(\frac{p(x)}{q(x)} \right)$$

where we use the convention $0 \ \log_2(\frac{0}{0}) = 0$, $0 \ \log_2(\frac{0}{q}) = \infty$ and $p \ \log_2(\frac{p}{0}) = \infty$ whenever $p, q > 0$.

2.1.1 Important theorems

Below, two important theorems for the modelling are stated and proved.

Lemma 1 (log sum inequality)**.** *Let $a_1, a_2, \ldots, a_n$ and $b_1, b_2, \ldots, b_n$ be non-negative numbers then*

$$\sum_{i=1}^{n} a_i \ \log_2(\tfrac{a_i}{b_i}) \geq \left(\sum_{i=1}^{n} a_i \right) \log_2 \left(\frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i} \right)$$

with equality iff $\frac{a_i}{b_i}$ is constant for all i. Again, the convention $0 \ \log_2(0) = 0$, $a \ \log_2(\frac{a}{0}) = \infty$ if $a > 0$ and $0 \ \log_2(\frac{0}{0}) = 0$ is used.

Theorem 1 (Properties of Kulback-Leibler divergence)**.** *Let p and q be two probability mass functions defined on $\mathfrak{X} = \{x_1, \ldots, x_n\}$. Then the following holds*

1. *$D(p||q) \geq 0$ with equality iff $p = q$*
2. *$D(p||q)$ is convex in the pair (p, q), i.e. if (p_1, q_1) and (p_2, q_2) are two pairs of probability mass functions, then*

$$D(\lambda p_1 + (1-\lambda)p_2||\lambda q_1 + (1-\lambda)q_2) \leq \lambda D(p_1||q_1) + (1-\lambda)D(p_2||q_2) \quad \forall\ 0 \leq \lambda \leq 1$$

Proof. For both parts of the proof the log sum inequality can be used.

$$\begin{aligned} D(p||q) &= \sum_{x\in\mathfrak{X}} p(x) \log_2 \left(\frac{p(x)}{q(x)}\right) \\ &\geq \left(\sum_{x\in\mathfrak{X}} p(x)\right) \log_2 \left(\frac{\sum_{x\in\mathfrak{X}} p(x)}{\sum_{x\in\mathfrak{X}} q(x)}\right) \\ &= 1\ \log_2 \left(\frac{1}{1}\right) = 0 \end{aligned}$$

with equality, iff $\frac{p(x)}{q(x)} = c$ for all $x \in \mathfrak{X}$. Since $p(x)$ and $q(x)$ both sum up to 1, $c = 1$ and therefore $p = q$.
For the second part we apply the log sum inequality to the term

$$\begin{aligned} \lambda p_1(x) \log_2 \left(\frac{\lambda p_1(x)}{\lambda q_1(x)}\right) + (1-\lambda)p_2(x) \log_2 \left(\frac{(1-\lambda)p_2(x)}{(1-\lambda)q_2(x)}\right) \geq \\ (\lambda p_1(x) + (1-\lambda)p_2(x)) \log_2 \left(\frac{\lambda p_1(x) + (1-\lambda)p_2(x)}{\lambda q_1(x) + (1-\lambda)q_2(x)}\right) \end{aligned}$$

for all $x \in \mathfrak{X}$. The second claim of the theorem follows by summing over all $x \in \mathfrak{X}$.

□

Theorem 2 (Properties of entropy)**.** *Let X be a discrete random variable with possible values in $\mathfrak{X} = \{x_1, \ldots, x_n\}$ and the probability mass function p, then, the following holds*

1. *$H(p) \geq 0$*
2. *$H(p)$ is concave*
3. *$H(p)$ is maximal for the uniform distribution*

Proof. Point one follows immediately from $0 \leq p(x) \leq 1$, which implies that $\log_2(\frac{1}{p(x)}) \geq 0$. For point two let the probability mass function u be the uniform distribution on $\mathfrak{X}$ and we get

$$\begin{aligned} D(p||u) &= \sum_{x \in \mathfrak{X}} p(x) \log_2 \left(\frac{p(x)}{u(x)} \right) \\ &= \sum_{x \in \mathfrak{X}} p(x) \log_2(p(x)) - \sum_{x \in \mathfrak{X}} p(x) \log_2(u(x)) \\ &= -H(p) + \log_2(n) \end{aligned} \tag{1}$$

and therefore $H(p) = \log_2(n) - D(p||u)$. The concavity of H follows then by the convexity of D.
For the third point we consider again equation (1), use that $D(p||u) \geq 0$ with equality iff $p = u$ and get

$$H(p) \leq \log_2(n) = H(u).$$

□

The first point of theorem 1 motivates the use of the Kullback-Leibler divergence as a measure of distance between two probability distributions. It does not define a metric on the space of probability distributions on a finite set $\mathfrak{X}$ though, since it is neither symmetric nor satisfies the triangle inequality. In spite of this fact, the term distance between p and q will be used for $D(p||q)$.

2.2 Assumptions

In the following, the term product will be used for a type of good, for example, a bottle of wine of a specific brand. The term article will be used for the different types of how a product is stored, for example, a bottle of wine can be stored on its own or in a carton box with 6 bottles. The individual units of an article will be called SKUs (stock keeping units) (cf. Ghiani et al. 2005). So from now on we are also talking about article correlation rather than product correlation. For the model that gets presented, the following assumptions are required:

1. For each article in the warehouse we have a measure of how much space is needed to store this article. Henceforth, this measure will be called article capacity and its units capacity units. If we store, for example, beverages we could use the filling capacity as a measure. A bottle of wine could therefore have an article capacity of 0.75 capacity units, whereas a carton box of wine could have an article capacity of 4.5 capacity units.

2. Each article can be stored in any storage location together with any other article, which is especially important if it is necessary to stack articles.

3. Each storage location in the warehouse has the same capacity, which is also given in capacity-units.

4. The warehouse has exactly one depot, where the articles that get commissioned have to be transported.

2.3 Model formulation

For the formulation of the model, the notations in table 1 are used.

notation	definition
n	number of articles
m	number of storage locations
α	parameter to control the impact of article correlation
β	parameter to control the impact of article turnover
γ	parameter to control the clustering of articles
λ_{ij}	percentage of orders in which article i and j are ordered together
μ_i	percentage of the turnover of article i of the aggregate turnover
d_k	shortest distance from storage location k to the depot
g_i	article-capacity of article i multiplied with the average inventory level of article i
L	storage location capacity
x^i	probability distribution of article i
M_ϵ	set of admissible probability distributions

Table 1: Model Notations

First we introduce a parameter $0 < \epsilon \leq \frac{1}{m}$ and define the set M_ϵ by

$$M_\epsilon = \{x \in \mathbb{R}^m : \epsilon \leq x_j \leq 1 \text{ and } \sum_{j=1}^{m} x_j = 1\}$$

, where we use subscripts for the entries of a vector. The parameter ϵ is important since we want to ensure that every article has the chance to be assigned to every storage location. If we let $0 \leq x^i_j \leq 1$, the storage locations which correspond to the support of x^i could become full and article i cannot be assigned to a free storage location anymore. So, by introducing ϵ as a lower bound for the probabilities, such situations can be prevented. Note that from now on x rather than p is used for a probability distribution, since the modelling results in a minimization problem where the variables are usually denoted by x. This is also the reason for requiring ϵ to be less or equal to $1/m$, since the uniform distribution for every article should later on be used as a starting point for a minimization algorithm.

Now, the aim is to measure the quality of a given set of article probability distributions $\{x^i \in M_\epsilon : i \in \{1, 2, \ldots, n\}\}$ with respect to the three factors mentioned in the intro-

duction with a function $F : M_\epsilon^n \to \mathbb{R}$. A low function value of F should correspond to a good quality, the higher it gets the lower the consideration of the important factors.
For the article correlation the Kullback-Leibler divergence can be used to measure the distance of the distributions x^i and x^j where the parameter λ_{ij} can be used as a weight. So, the first term in the function F will be

$$f_1(x^1, \dots, x^n) = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \lambda_{ij} D(x^i || x^j)$$

. One observes, that the higher the value of λ_{ij} the higher the impact of $D(x^i||x^j)$, or, in other words, the higher the value of λ_{ij} the lower should the distance between x^i and x^j be to result in a low function value of f_1.
For the article turnover, the parameters d_k and the weights μ_i are used to quantify the chances for an article to be assigned to a storage location near the depot. The second part of the function F will therefore be

$$f_2(x^1, \dots, x^n) = \sum_{i=1}^{n} \mu_i \sum_{k=1}^{m} x_k^i d_k$$

. Note that a low function value of f_2 means, that articles with a higher sale rate are assigned more likely to storage locations near the depot.
The last term should measure how the articles are distributed in the warehouse. If we, for example, use the uniform distribution for x^i with $i \in \{1, 2, \dots, n\}$, the article i will be assigned to every storage location with equal probability and can therefore be found in every storage location with equal probability later on. As mentioned in the introduction, this could lead to an increase in picking time and should therefore be penalized. By theorem 2 point three the entropy is maximized by the uniform distribution, and one can easily check that it is minimized by probability distributions, which are concentrated in a single point. It is therefore natural to introduce an entropy penalty term

$$f_3(x^1, \dots, x^n) = \sum_{i=1}^{n} H(x^i)$$

to encourage article clustering in the warehouse. The function F gets defined as

$$F = \alpha f_1 + \beta f_2 + \gamma f_3$$

, where the non-negative parameters α,β and γ are used to control the impact of each factor measured by the functions f_1,f_2 and f_3, since by theorem 1 and 2 both, the entropy and the Kullback-Leibler divergence, are non-negative. With this tool to measure the consideration of the three factors article correlation, article turnover, and article clustering at our disposal, the aim is to find a set of admissible probability distributions

for each article with the lowest function value of F. Without additional constraints this would lead to probability distributions, which are all concentrated in the storage location nearest to the depot. This is obviously due to the fact that the capacity constraints of the storage locations has not been considered yet. The parameters g^i are the amount of capacity units needed to store the average inventory level of article i, so, for a given set of probability distributions $\{x^i \in M_\epsilon : i \in \{1, 2, \dots, n\}\}$, it should hold

$$\sum_{i=1}^{n} g_i x_k^i \leq L \quad \forall k \in \{1, 2, \dots, m\}$$

, which says that the expected value of capacity units stored at storage location k is smaller than the storage location capacity L for all k. If we again require that it should be admissible to choose the uniform distribution for all articles, this leads to the assumption

$$\sum_{i=1}^{n} g_i \leq mL$$

. Since this inequality means that the aggregate average inventory level is smaller than the warehouse capacity, this is quite a natural assumption. The final formulation of the model is therefore

$$\begin{aligned} \min_{(x^1,\dots,x^n)\in M_\epsilon^n} \alpha \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \lambda_{ij} D(x^i||x^j) + \beta \sum_{i=1}^{n} \mu_i \sum_{k=1}^{m} x_k^i d_k + \gamma \sum_{i=1}^{n} H(x^i) \\ \text{s.t } \sum_{i=1}^{n} g_i x_k^i \leq L \quad \forall k \in \{1, 2, \dots, m\}. \end{aligned}$$

2.4 Further investigation of the model

If we now also incorporate the constraints of M_ϵ^n in the formulation of the model, we have

$$\min_{(x^1,\dots,x^n)\in \mathbb{R}^{nm}} \quad \alpha \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \lambda_{ij} D(x^i||x^j) + \beta \sum_{i=1}^{n} \mu_i \sum_{k=1}^{m} x_k^i d_k + \gamma \sum_{i=1}^{n} H(x^i)$$

$$\begin{aligned} & (i) \quad \epsilon \leq x_j^i \leq 1 \quad \forall i \in \{1, \dots, n\} \text{ and } j \in \{1, \dots, m\} \\ \text{s.t.} \quad & (ii) \quad \sum_{k=1}^{m} x_k^i = 1 \quad \forall i \in \{1, \dots, n\} \\ & (iii) \quad \sum_{i=1}^{n} g_i x_k^i \leq L \quad \forall k \in \{1, \dots, m\} \end{aligned}$$

. First of all, the inequality constraints (iii) can be turned into equality constraints by introducing slack variables $0 \leq s_k \leq \infty$. If we again use subscripts for the entries of a

vector s the minimization problem becomes

$$\min_{(x^1,\dots,x^n,s)\in\mathbb{R}^{nm+m}} \quad \alpha \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} \lambda_{ij} D(x^i||x^j) + \beta \sum_{i=1}^{n}\mu_i \sum_{k=1}^{m} x_k^i d_k + \gamma \sum_{i=1}^{n} H(x^i)$$

$$\text{s.t.} \quad \begin{aligned} &(i) && \epsilon \le x_j^i \le 1 && \forall i \in \{1,\dots,n\} \text{ and } j \in \{1,\dots,m\} \\ &(ii) && \sum_{k=1}^{m} x_k^i = 1 && \forall i \in \{1,\dots,n\} \\ &(iii) && \sum_{i=1}^{n} g_i x_k^i + s_k = L && \forall k \in \{1,\dots,m\} \\ &(iv) && 0 \le s_k \le \infty && \forall k \in \{1,\dots,m\} \end{aligned}$$

. Since the equality constraints in (ii) and (iii) are linear, they can be written in matrix form and together with the definitions

$$\mathrm{C} \coloneqq \begin{pmatrix} \mathbb{1}_m^\intercal & \dots & \dots & \dots & 0_m^\intercal \\ \vdots & \mathbb{1}_m^\intercal & \dots & \dots & \vdots \\ \vdots & \dots & \dots & \mathbb{1}_m^\intercal & 0_m^\intercal \\ g_1\mathbb{I}_m & \dots & \dots & g_n\mathbb{I}_m & \mathbb{I}_m \end{pmatrix} \in \mathbb{R}^{(n+m)\times(nm+m)} \qquad \mathrm{b} \coloneqq \begin{pmatrix} \mathbb{1}_n \\ L\cdot\mathbb{1}_m \end{pmatrix} \in \mathbb{R}^{n+m}$$

$$\mathrm{l} \coloneqq \begin{pmatrix} \epsilon\cdot\mathbb{1}_{nm} \\ 0_m \end{pmatrix} \in \mathbb{R}^{nm+m} \qquad\qquad \mathrm{u} \coloneqq \begin{pmatrix} \mathbb{1}_{nm} \\ \infty\cdot\mathbb{1}_m \end{pmatrix} \in \mathbb{R}^{nm+m}$$

, where $\mathbb{1}_m$ is the column vector of ones in $\mathbb{R}^m$, 0_m is the column vector of zeros in $\mathbb{R}^m$ and $\mathbb{I}_m$ is the identity matrix in $\mathbb{R}^{m\times m}$, the model finally has the form of a bound-constrained minimization problem with linear equality constraints

$$\min_{(x^1,\dots,x^n,s)\in\mathbb{R}^{nm+m}} \quad F(x^1,\dots,x^n)$$

$$\text{s.t.} \quad \begin{aligned} &(i) && \mathrm{l} \le \left(x^1,\dots,x^n,s\right)^\intercal \le \mathrm{u} \\ &(ii) && \mathrm{C}\left(x^1,\dots,x^n,s\right)^\intercal = \mathrm{b} \end{aligned}$$

. Note that the bound-constraints have to be understood componentwise and the infinite entries in u denote the components, which are unconstrained from above. Now, before going into detail on how to solve these kind of problems, we will have a look on a few properties of the model. First of all, we cannot hope for a convex problem in general, since with f_3 we also have a concave component in the function F. So, depending on

the choice of α and γ, we will possibly have to deal with negative curvature. Secondly, the parameter ϵ has the positive side effect that F is twice continuously differentiable inside the admissible set of variables. By writing f_1 in the form

$$\begin{aligned}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\lambda_{ij}D(x^i||x^j) &= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=1}^{m}x_k^i\log_2(x_k^i)-x_k^i\log_2(x_k^j)\\ &= \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=1}^{m}x_k^i\log_2(x_k^i)-\sum_{i=1}^{s-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=1}^{t}x_k^i\log_2(x_k^j)\\ &-\sum_{i=1}^{s-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=t+1}^{m}x_k^i\log_2(x_k^j)-\sum_{i=s}^{n-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=1}^{t}x_k^i\log_2(x_k^j)\\ &-\sum_{i=s}^{n-1}\sum_{j=i+1}^{n}\lambda_{ij}\sum_{k=t+1}^{m}x_k^i\log_2(x_k^j)\end{aligned}$$

for $s\in\{1,\dots,n\}$ and $t\in\{1,\dots,m\}$, with the convention that empty sums are zero, one can calculate

$$\begin{aligned}\frac{\partial f_1}{\partial x_t^s} &= \sum_{j=s+1}^{n}\lambda_{sj}\left(\log_2(x_t^s)+\frac{1}{\ln(2)}\right)-\sum_{i=1}^{s-1}\lambda_{is}\frac{x_t^i}{\ln(2)x_t^s}-\sum_{j=s+1}^{n}\lambda_{sj}\log_2(x_t^j)\\ &= \sum_{j=s+1}^{n}\lambda_{sj}\left(\log_2(x_t^s)-\log_2(x_t^j)+\frac{1}{\ln(2)}\right)-\sum_{i=1}^{s-1}\lambda_{is}\frac{x_t^i}{\ln(2)x_t^s}\end{aligned}$$

and with $u\in\{1,\dots,n\}$ and $w\in\{1,\dots,m\}$ the second partial derivatives are

$$\begin{aligned}\frac{\partial^2 f_1}{\partial x_t^s\partial x_w^u} &= 0 \quad \text{for } w\neq t\\ \frac{\partial^2 f_1}{\partial^2 x_t^s} &= \sum_{j=s+1}^{n}\frac{\lambda_{sj}}{\ln(2)x_t^s}+\sum_{i=1}^{s-1}\frac{\lambda_{is}x_t^i}{\ln(2)x_t^{s2}}\\ \frac{\partial^2 f_1}{\partial x_t^s\partial x_t^u} &= \begin{cases}-\frac{\lambda_{us}}{\ln(2)x_t^s} & u<s\\ -\frac{\lambda_{su}}{\ln(2)x_t^u} & u>s\end{cases}\end{aligned}$$

. By defining

$$\begin{aligned}D_s &:= \sum_{j=s+1}^{n}\frac{\lambda_{sj}}{\ln(2)}\mathrm{diag}\left(\frac{1}{x^s}\right)+\sum_{i=1}^{s-1}\frac{\lambda_{is}}{\ln(2)}\mathrm{diag}\left(\frac{x^i}{x^{s2}}\right)\in\mathbb{R}^{m\times m}\\ N_s &:= -\frac{1}{\ln(2)}\mathrm{diag}\left(\frac{1}{x^s}\right)\in\mathbb{R}^{m\times m}\end{aligned}$$

, where $\mathrm{diag}(a)$ for a vector $a \in \mathbb{R}^N$ denotes the diagonal matrix in $\mathbb{R}^{N\times N}$ with the diagonal entries given by a and $\frac{1}{a}$, respectively a^2, are defined componentwise, the hessian of f_1 can be written in the following form

$$\nabla^2 f_1 = \begin{pmatrix} D_1 & \lambda_{12}N_2 & \lambda_{13}N_3 & \dots & \lambda_{1n}N_n \\ \lambda_{12}N_2 & D_2 & \lambda_{23}N_3 & \dots & \vdots \\ \lambda_{13}N_3 & \lambda_{23}N_3 & \ddots & \dots & \vdots \\ \vdots & \dots & \dots & \ddots & \vdots \\ \lambda_{1n}N_n & \dots & \dots & \dots & D_n \end{pmatrix}$$

. With the definitions

$$I_i = \begin{pmatrix} 0_{m\times m} & & & & \\ & \ddots & & & \\ & & \mathbb{I}_m & & \\ & & & \ddots & \\ & & & & 0_{m\times m} \end{pmatrix} \in \mathbb{R}^{nm\times nm} \quad \text{with} \quad 0_{m\times m} = \begin{pmatrix} 0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 0 \end{pmatrix} \in \mathbb{R}^{m\times m}$$

$$d = (\underbrace{d_1, \dots, d_m, d_1, \dots, d_m, \dots, d_1, \dots, d_m}_{\text{n-times}})^\intercal \in \mathbb{R}^{nm}$$

$$v = (x^1, \dots, x^n)^\intercal \in \mathbb{R}^{nm}$$

$$\mathcal{M}_i = \mu_i I_i$$

, where the block $\mathbb{I}_m$ in I_i goes from row and column $(i-1)m+1$ to row and column im, f_2 can be written as

$$\sum_{i=1}^{n} \mu_i \sum_{k=1}^{m} x_k^i d_k = \sum_{i=1}^{n} \mu_i \langle I_i v, d\rangle = \sum_{i=1}^{n} \langle \mathcal{M}_i v, d\rangle$$

and one can calculate for f_2 and f_3

$$\begin{aligned}\nabla f_2 &= \sum_{i=1}^{n} \mathcal{M}_i^{\intercal} d \\ \nabla^2 f_2 &= 0 \\ \nabla f_3 &= -\log_2(v) - \frac{1}{\ln(2)} \mathbb{1}_{nm} \\ \nabla^2 f_3 &= -\frac{1}{\ln(2)} \operatorname{diag}\left(\frac{1}{v}\right)\end{aligned}$$

, where again $\log_2(v)$ is defined componentwise. So, finally, the hessian of F has the form

$$\nabla^2 F = \begin{pmatrix} \alpha D_1 + \gamma N_1 & \alpha\lambda_{12} N_2 & \alpha\lambda_{13} N_3 & \dots & \alpha\lambda_{1n} N_n \\ \alpha\lambda_{12} N_2 & \alpha D_2 + \gamma N_2 & \alpha\lambda_{23} N_3 & \dots & \vdots \\ \alpha\lambda_{13} N_3 & \alpha\lambda_{23} N_3 & \ddots & \dots & \vdots \\ \vdots & \dots & \dots & \ddots & \vdots \\ \alpha\lambda_{1n} N_n & \dots & \dots & \dots & \alpha D_n + \gamma N_n \end{pmatrix}$$

. At worst, the sparsity of this matrix is given by $(m-1)/m$. Due to this fact and its simple form, exact second order derivatives will be used in my implementation.

3 Theory

The aim of this chapter is to build up the theory of a trust region based gradient projection method for equality- and bound-constrained nonlinear programs. The presented algorithm is a reduced version of the one, used in the software package LANCELOT, which is designed for solving large-scale nonlinear optimization problems (see Conn et al. 2010). In chapter 3.1 some general results of constrained optimization are summarized. In chapter 3.2 penalty methods, especially the quadratic penalty method and the augmented Lagrangian method, will be presented. Afterwards, the special case of the bound-constrained Lagrangian method will be discussed in chapter 3.3, while chapter 3.4 focuses on the gradient projection method. For the most part, the presented results follow the ones in Nocedal et Wright (2006), if not stated otherwise.

3.1 Fundamentals of constrained optimization

To begin with, we have a look at the general constrained minimization problem.

$$\min_{x \in \mathbb{R}^N} F(x) \quad \text{subject to} \begin{cases} c_i(x) = 0 & i \in \mathcal{E} \\ c_i(x) \geq 0 & i \in \mathcal{I} \end{cases} \tag{2}$$

, with $F : \mathbb{R}^N \to \mathbb{R}$ and $c_i : \mathbb{R}^N \to \mathbb{R}$. The c_i with $i \in \mathcal{E}$ are the equality constraints and the c_i with $i \in \mathcal{I}$ are the inequality constraints. The feasible set Ω is then defined by

$$\Omega = \{x \in \mathbb{R}^N : c_i(x) = 0 \ , \ i \in \mathcal{E} \ , \ c_i(x) \geq 0 \ , \ i \in \mathcal{I}\}$$

. For any feasible $x \in \Omega$ the active set $\mathcal{A}(x)$ is defined by

$$\mathcal{A}(x) = \mathcal{E} \cup \{i \in \mathcal{I} : c_i(x) = 0\}$$

and we say that the linear independence constraint qualification (LICQ) holds at x if the set of active constraint gradients $\{\nabla c_i(x) \ , \ i \in \mathcal{A}(x)\}$ is linearly independent. To state first and second order necessary conditions we also need the definition of the Lagrangian function of problem (2):

$$\mathcal{L}(x, \bar{\lambda}) = F(x) - \sum_{i \in \mathcal{E} \cup \mathcal{I}} \bar{\lambda}_i c_i(x)$$

Note that $\bar{\lambda}$ is used for the Lagrange multipliers in order to avoid confusion with the model parameter.

3.1.1 First order necessary conditions

To state first order conditions we need the objective function F and the constraint functions c_i to be continuously differentiable. The following results give conditions on a local solution of problem (2) under additional assumptions, the so-called first order necessary conditions.

Suppose that x^* is a local solution of problem (2) and that the LICQ holds at x^*, then there exists a Lagrange multiplier vector $\bar{\lambda}^*$ with components $\bar{\lambda_i}^*$, such that the following conditions hold

$$\begin{array}{lccll} (i) & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*) & = & 0 & \\ (ii) & c_i(x^*) & = & 0 & \forall i \in \mathcal{E} \\ (iii) & c_i(x^*) & \geq & 0 & \forall i \in \mathcal{I} \\ (iv) & \bar{\lambda_i}^* & \geq & 0 & \forall i \in \mathcal{I} \\ (v) & \bar{\lambda_i}^* c_i(x^*) & = & 0 & \forall i \in \mathcal{E} \cup \mathcal{I} \end{array}$$

These conditions are also known as the Karush-Kuhn-Tucker, or KKT conditions. The point x^* is called Karush-Kuhn-Tucker point, or KKT point. The conditions (v) are called complementary conditions, which say that for all $i \in \mathcal{E} \cup \mathcal{I}$ either c_i is active, or $\bar{\lambda_i}^* = 0$, or both. If exactly one of them is zero for each index $i \in \mathcal{I}$, we say that the strict complementary condition holds.

3.1.2 Second order necessary conditions

To state the second order necessary conditions we first suppose that the objective function F and the constraint functions c_i are twice continuously differentiable. Secondly, we need two more definitions. The first one is the set of linearized feasible directions $\mathcal{F}(x)$ for a feasible point $x \in \Omega$, which is defined as

$$\mathcal{F}(x) = \{d \in \mathbb{R}^N : d^\intercal \nabla c_i(x) = 0 \ , \ \forall i \in \mathcal{E} \ , \ d^\intercal \nabla c_i(x) \geq 0 \ , \ \forall i \in \mathcal{A}(x) \cap \mathcal{I}\}$$

. For the second one suppose that x^* is a KKT point with Lagrange multiplier vector $\bar{\lambda}^*$. The critical cone $\mathcal{C}(x^*, \bar{\lambda}^*)$ is defined as

$$\mathcal{C}(x^*, \bar{\lambda}^*) = \{w \in \mathcal{F}(x^*) : \nabla c_i(x^*)^\intercal w = 0, \text{ for all } i \in \mathcal{A}(x^*) \cap \mathcal{I} \text{ with } \bar{\lambda_i}^* > 0\}$$

. The second order necessary conditions are then given by the following:

Suppose that x^* is a local solution of (2) at which the LICQ holds and that $\bar{\lambda}^*$ is the Lagrange multiplier vector for which the KKT conditions are satisfied. Then it holds

$$w^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w \geq 0 \quad \forall w \in \mathcal{C}(x^*, \bar{\lambda}^*).$$

3.1.3 Second order sufficient conditions

Until now we have only stated necessary conditions for a local solution of (2). The next step will be to state conditions for a point x^* that are sufficient for being a local solution.

Suppose that for a feasible point x^* there exists a Lagrange multiplier vector $\bar{\lambda}^*$ such that the first order necessary conditions are fulfilled. Further suppose that

$$w^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w > 0 \quad \forall w \in \mathcal{C}(x^*, \bar{\lambda}^*) \setminus \{0\}$$

. Then x^* is a strict local solution of (2).

Note that the LICQ is not required and that the conditions stated above are very similar to the necessary conditions, with the exception of the positive semidefiniteness of $\nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)$ in $\mathcal{C}(x^*, \bar{\lambda}^*)$ being replaced by a strict positive definiteness. With regard to the model, the aim is now to find a KKT point and check if the second order sufficient conditions (SOSC) are satisfied. This will be done numerically in the implementation. To do so, we follow (Kelly et Kupferschmid 1998) and note first of all that $\mathcal{C}(x^*, \bar{\lambda}^*)$ has, in the case that the LICQ holds at x^* and that strict complementary holds, the form

$$\mathcal{C}(x^*, \bar{\lambda}^*) = \text{Null}\left((\nabla c_i(x^*)^\intercal)_{i \in \mathcal{A}(x^*)}\right) \tag{3}$$

. Now, let $\mathcal{N} \in \mathbb{R}^{N \times r}$ be the matrix, whose columns make up a basis for (3), then every $w \in \mathcal{C}(x^*, \bar{\lambda}^*)$ has the form $w = Ny$ for some $y \in \mathbb{R}^r$ and one can calculate

$$w^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w = z + \sum_{i=1}^{r} y_i^2 \mathcal{N}^{i\intercal} \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) \mathcal{N}^i$$

, where $\mathcal{N}^i$ denotes the ith column of $\mathcal{N}$ and z collects all terms of the form

$$y_i y_j \mathcal{N}^{i\intercal} \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) \mathcal{N}^j \text{ for } i \neq j$$

. If we now suppose that the vectors $\mathcal{N}^i$ are $\nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)$-conjugate, the SOSC are fulfilled if

$$\mathcal{N}^{i\intercal} \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) \mathcal{N}^i > 0 \quad \forall i \in \{1, \ldots, r\}$$

. Checking the SOSC therefore reduces to rendering the columns of $\mathcal{N}$ conjugate with

respect to $\nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)$, which can be accomplished by a modified Gram Schmidt procedure. We therefore define

$$G_S(x_k) = x_k - \sum_{j=1}^{k-1} \frac{\langle x_j, \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)x_k\rangle}{\langle x_k, \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)x_k\rangle} x_j$$
$$u_k = G_S(\mathcal{N}^k)$$

and check $u_j^\intercal \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)u_j$ for each $j \in \{1, \ldots, r\}$. If one of them fails to be positive the algorithm can stop, otherwise the SOSC are satisfied.

3.1.4 Combinatorial aspects of inequality constraints

The main challenge in nonlinear programming is dealing with inequality constraints, especially in deciding which of the inequality constraints are active. The approach of active set methods is to guess the set of constraints which are active at the solution. This set is called the working set and gets denoted by $\mathcal{W}$. Then, the constraints which are not in the working set are ignored, the constraints in the working set are replaced with equality constraints, and one checks if there is a Lagrange multiplier vector such that the KKT conditions of the problem are satisfied. This is due to the fact that equality constrained problems are easier to solve than inequality constrained problems. Since the number of possible working sets can get very high, this aspect is referred to as the combinatorial difficulty of nonlinear programming. The algorithm that gets presented in chapter 3.4 is an active set method that benefits from a good choice of the working set.

3.2 Penalty methods

In this chapter we focus on penalty methods. As the name already indicates, rather than strictly requiring feasibility, these methods penalise constraint violations by an additional penalty term in the objective function, which is zero if the constraints are met and greater than zero otherwise. The idea behind this approach is to use algorithms of unconstrained optimization to solve constrained problems. For the rest of this chapter we focus on the general equality-constrained problem

$$\min_{x \in \mathbb{R}^N} F(x) \quad \text{subject to } c_i(x) = 0 \quad \forall i \in \mathcal{E} \tag{4}$$

, where the objective function F and the constraint functions c_i are supposed to be twice continuously differentiable.

3.2.1 Quadratic penalty method

A very simple penalty function is the quadratic penalty function, which is defined as

$$\mathcal{Q}(x;\theta) = F(x) + \frac{\theta}{2}\sum_{i\in\mathcal{E}} c_i^2(x)$$

, where θ is the so called penalty parameter. It is quite intuitive to consider a sequence of $(\theta_k)_{k\in\mathbb{N}}$ with $\theta_k \to \infty$ for $k \to \infty$ and to find approximate minimizers x^k of $\mathcal{Q}(x;\theta_k)$ for each k, since this way constraint violations get penalized with increasing severity. Because of the smoothness of the penalty term, unconstrained optimization techniques can be used and previous solutions can be reused in the choice of the next starting point. A general framework of the Quadratic penalty method can be specified as (see Nocedal et Wright 2006 [p.501])

Data: $\theta_0 > 0$, $(\tau_k)_{k\in\mathbb{N}}$ with $\tau_k \to 0$, starting point x_s^0
Result: approximate solution x^*
for $k = 0, 1, 2, \ldots$ **do**
  Find approximate minimizer x^k of $\mathcal{Q}(\cdot;\theta_k)$
  starting at x_s^k with $||\nabla_x \mathcal{Q}(x^k;\theta_k)|| \leq \tau_k$;
  if *convergence test satisfied* **then**
    return $x^* = x^k$;
  else
    Set new penalty parameter $\theta_{k+1} > \theta_k$;
    Set new starting point x_s^{k+1};
  end
end

Algorithm 1: Quadratic Penalty Method

. The convergence properties of this method are described in the following two theorems.

Theorem 3. *Suppose that each x^k in algorithm 1 is the exact global minimizer of $\mathcal{Q}(\cdot;\theta_k)$ and that $\theta_k \to \infty$ for $k \to \infty$, then every accumulation point x^* of $(x^k)_{k\in\mathbb{N}}$ is a global solution of problem (4).*

Proof. Let $\bar{x}$ be a global solution of problem (4), i.e. $F(\bar{x}) \leq F(x)$ for all x with $c_i(x) = 0$ $\forall i \in \mathcal{E}$. For every accumulation point x^* there exists a subsequence $(x^{k_l})_{l \in \mathbb{N}}$ with

$$\lim_{l \to \infty} x^{k_l} = x^*$$

. Since all the x^{k_l} are exact minimizers of $\mathcal{Q}(\cdot; \theta_{k_l})$, we have that

$$F(x^{k_l}) + \frac{\theta_{k_l}}{2} \sum_{i \in \mathcal{E}} c_i^2(x^{k_l}) \leq F(\bar{x}) + \frac{\theta_k}{2} \sum_{i \in \mathcal{E}} c_i^2(\bar{x}) = F(\bar{x}) \tag{5}$$

and therefore

$$\sum_{i \in \mathcal{E}} c_i^2(x^{k_l}) \leq \frac{2}{\theta_{k_l}} \left(F(\bar{x}) - F(x^{k_l}) \right) \tag{6}$$

. If we now take the limit $l \to \infty$ on both sides of (6) we get

$$\sum_{i \in \mathcal{E}} c_i^2(x^*) \leq 0$$

and therefore that x^* is feasible. Due to the nonnegativity of θ_{k_l} and $c_i^2(x^{k_l})$ equation (5) gives us that $F(x^{k_l}) \leq F(\bar{x})$ and by taking the limit $l \to \infty$ again this results in $F(x^*) \leq F(\bar{x})$, which concludes the proof, since $\bar{x}$ is a global solution of problem (4) and x^* has an objective function value lower or equal, and therefore equal, to $F(\bar{x})$. □

The next theorem concerns the more realistic case in which the subproblems $\min_{x \in \mathbb{R}^N} \mathcal{Q}(x; \theta_k)$ are only solved approximately, but with increasing accuracy. Note that in the general framework of algorithm 1 it is not certain, that the condition $||\nabla_x \mathcal{Q}(x^k; \theta_k)|| \leq \tau_k$ can be satisfied in every step, since a too low penalty parameter θ_k can cause the iterates to move away from the feasible region. The next theorem therefore assumes, that the stopping criteria is fulfilled in every step, which is a quite optimistic assumption due to Nocedal et Wright. It also gives an estimate of the Lagrange multiplier vector, which will be used later on in the augmented Lagrangian method.

Theorem 4. *Suppose that in the framework of algorithm 1 the condition* $||\nabla_x \mathcal{Q}(x^k;\theta_k)|| \leq \tau_k$ *holds for all* k *and that* $\theta_k \to \infty$ *for* $k \to \infty$*. Then every accumulation point* x^* *of the sequence* $(x^k)_{k\in\mathbb{N}}$ *is a stationary point of the function* $||c(x)||^2$ *if it is infeasible or a KKT point of problem (4) if the constraint gradients* $\nabla c_i(x^*)$ *are linearly independent. In the later case we also have that*

$$\lim_{l\to\infty} -\theta_{k_l} c_i(x^{k_l}) = \bar{\lambda_i}^* \quad \forall i \in \mathcal{E} \tag{7}$$

, where the subsequence $(x^{k_l})_{l\in\mathbb{N}}$ *is such that* $x^{k_l} \to x^*$ *for* $l \to \infty$ *and* $\bar{\lambda}^*$ *is the Lagrange multiplier vector corresponding to* x^**.*

Proof. Let x^* be an accumulation point of $(x^k)_{k\in\mathbb{N}}$ and the subsequence $(x^{k_l})_{l\in\mathbb{N}}$ such that $x^{k_l} \to x^*$ for $l \to \infty$. One can calculate that

$$\nabla_x \mathcal{Q}(x^{k_l};\theta_{k_l}) = \nabla F(x^{k_l}) + \sum_{i\in\mathcal{E}} \theta_{k_l} c_i(x^{k_l}) \nabla c_i(x^{k_l}) \tag{8}$$

and then by assumption it holds that

$$||\nabla F(x^{k_l}) + \sum_{i\in\mathcal{E}} \theta_{k_l} c_i(x^{k_l}) \nabla c_i(x^{k_l})|| \leq \tau_{k_l} \tag{9}$$

. By using the reverse triangle inequality on (9) we get

$$||\sum_{i\in\mathcal{E}} c_i(x^{k_l}) \nabla c_i(x^{k_l})|| \leq \frac{1}{\theta_{k_l}} \left(\tau_{k_l} + ||\nabla F(x^{k_l})||\right)$$

and by taking the limit $l \to \infty$ this results in

$$\sum_{i\in\mathcal{E}} c_i(x^*) \nabla c_i(x^*) = 0 \tag{10}$$

. If there is an $j \in \mathcal{E}$ such that $c_j(x^*) \neq 0$, which means that x^* is infeasible and the constraint gradients are linearly dependent, equation (10) implies that the accumulation point x^* is a stationary point of $||c(x)||^2$. On the other hand, if the constraint gradients are linearly independent, equation (10) implies that x^* is feasible. What is left to proof

is that x^* is indeed a KKT point of problem (4) and that (7) holds. To do so, we define

$$A(x)^\intercal = (\nabla c_i(x))_{i\in\mathcal{E}}$$
$$\bar{\lambda}^k = -\theta_k c(x^k)$$

. Now equation (8) implies that

$$A(x^{k_l})^\intercal \bar{\lambda}^{k_l} = \nabla F(x^{k_l}) - \nabla_x \mathcal{Q}(x^{k_l};\theta_{k_l}) \tag{11}$$

. Since $A(x^{k_l})^\intercal$ has full column rank for sufficiently large l, the matrix $A(x^{k_l})A(x^{k_l})^\intercal$ is regular and we can multiply equation (11) on both sides from the left with $A(x^{k_l})$ to get

$$\bar{\lambda}^{k_l} = \left(A(x^{k_l})A(x^{k_l})^\intercal\right)^{-1} A(x^{k_l})\left(\nabla F(x^{k_l}) - \nabla_x \mathcal{Q}(x^{k_l};\theta_{k_l})\right)$$

. By taking the limit $l \to \infty$ we get

$$\lim_{l\to\infty} \bar{\lambda}^{k_l} = \left(A(x^*)A(x^*)^\intercal\right)^{-1} A(x^*)\nabla F(x^*) =: \bar{\lambda}^*$$

, since $\nabla_x \mathcal{Q}(x^{k_l};\theta_{k_l})$ goes by assumption to zero. By taking again the limit $l \to \infty$ in equation (11) we also get

$$\nabla F(x^*) - A(x^*)^\intercal \bar{\lambda}^* = 0$$

, which completes the last two points. □

This theorem gives convergence properties of the quadratic penalty method but it also points out its weaknesses. The approximate minimizers x^k do not fulfill the feasibility condition $c_i(x^k) = 0$ for all $i \in \mathcal{E}$ but rather

$$c_i(x^k) \approx -\frac{\bar{\lambda_i}^*}{\theta_k} \quad \forall i \in \mathcal{E}$$

. If $\theta_k \to \infty$, we have that $c_i(x^k) \to 0$ but the question is, if this systematic errors can be avoided and if the ability of approximate minimizers to satisfy the feasibility constraints can be improved further, also for moderate values of θ_k. This can be accomplished by the augmented Lagrangian method.

3.2.2 Augmented Lagrangian method

The augmented Lagrangian for problem (4) can be seen as a mixture of the Lagrangian and the quadratic penalty function. It is defined as

$$\mathcal{L}_A(x, \bar{\lambda}; \theta) = F(x) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i c_i(x) + \frac{\theta}{2} \sum_{i \in \mathcal{E}} c_i^2(x)$$

. The idea is, to estimate the Lagrangian multiplier vector in every iteration and to fix the penalty parameter and this estimate momentarily to perform a minimization with sole respect to the variable x. To motivate the augmented Lagrangian method suppose we are in the k-th iteration and have an Lagrange multiplier estimate $\bar{\lambda}^k$ and a penalty parameter θ_k, as well as an approximate minimizer x^k of $\mathcal{L}_A(\cdot, \bar{\lambda}^k; \theta_k)$. Optimality conditions of unconstrained minimization then give that

$$0 \approx \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k) = \nabla F(x^k) - \sum_{i \in \mathcal{E}} \left(\bar{\lambda}_i^k - \theta_k c_i(x^k) \right) \nabla c_i(x^k) \tag{12}$$

. By comparing (12) with the first order necessary conditions of (4) we get

$$\bar{\lambda}_i^* \approx \bar{\lambda}_i^k - \theta_k c_i(x^k) \quad \forall i \in \mathcal{E}$$

, which suggests the multiplier update

$$\bar{\lambda}_i^{k+1} = \bar{\lambda}_i^k - \theta_k c_i(x^k) \quad \forall i \in \mathcal{E}$$

. Note the similarity to (7) and that we now have

$$c_i(x^k) \approx -\frac{1}{\theta_k} (\bar{\lambda}_i^* - \bar{\lambda}_i^k) \quad \forall i \in \mathcal{E}$$

, which means that a good estimate of $\bar{\lambda}^*$ results in an infeasibility of x^k which is now much smaller than in the quadratic penalty method. A general framework for the augmented Lagrangian method can now be specified as (see Nocedal et Wright 2006 [p.515]):

Data: $\theta_0 > 0$, $\tau_0 > 0$, starting point x_s^0, starting Lagrange multiplier $\bar{\lambda}^0$
Result: approximate solution x^*, approximate Lagrange multiplier $\bar{\lambda}^*$
for $k = 0, 1, 2, \ldots$ **do**
  Find approximate minimizer x^k of $\mathcal{L}_A(\cdot, \bar{\lambda}^k; \theta_k)$
  starting at x_s^k with $||\nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)|| \leq \tau_k$;
  if *convergence test satisfied* **then**
    Set $\bar{\lambda}_i^{k+1} = \bar{\lambda}_i^k - \theta_k c_i(x^k)$ for all $i \in \mathcal{E}$;
    return $x^* = x^k$ and $\bar{\lambda}^* = \bar{\lambda}^{k+1}$;
  else
    Set $\bar{\lambda}_i^{k+1} = \bar{\lambda}_i^k - \theta_k c_i(x^k)$ for all $i \in \mathcal{E}$;
    Set new penalty parameter $\theta_{k+1} \geq \theta_k$;
    Set new starting point x_s^{k+1};
    Set new tolerance τ_{k+1};
  end
end

Algorithm 2: Augmented Lagrangian method

. The tolerance τ_{k+1} can be chosen depending on the infeasibility of x^k. The penalty parameter θ_{k+1} may be increased, if the feasibility of x^k is insufficient, and the new starting point x_s^{k+1} is usually set to x^k. Again, the convergence properties of the augmented Lagrangian method will be discussed in two theorems.

Theorem 5. *Let x^* be a local solution of problem (4) at which the LICQ holds. Additionally, assume that the second order sufficient conditions hold for $(x^*, \bar{\lambda}^*)$, where $\bar{\lambda}^*$ is the Lagrange multiplier vector corresponding to x^*. Then, there is a threshold parameter $\bar{\theta}$, such that for all $\theta \geq \bar{\theta}$, x^* is a strict local minimizer of $\mathcal{L}_A(\cdot, \bar{\lambda}^*; \theta)$.*

Proof. To prove that x^* is a strict local minimizer of $\mathcal{L}_A(\cdot, \bar{\lambda}^*; \theta)$ for sufficiently large values of θ we show that

$$\begin{aligned} \nabla_x \mathcal{L}_A(x^*, \bar{\lambda}^*; \theta) &= 0 \\ \nabla_{xx}^2 \mathcal{L}_A(x^*, \bar{\lambda}^*; \theta) &> 0 \end{aligned} \tag{13}$$

for large values of θ. Since x^* is a local solution of problem (4) at which the LICQ holds the first order necessary conditions give that $\nabla_x \mathcal{L}(x^*, \bar{\lambda}^*) = 0$ and $c_i(x^*) = 0$ for all

$i \in \mathcal{E}$ and therefore we have

$$\begin{aligned}\nabla_x \mathcal{L}_A(x^*, \bar{\lambda}^*; \theta) &= \nabla F(x^*) - \sum_{i \in \mathcal{E}} \left(\bar{\lambda}_i^* - \theta c_i(x^*)\right) \nabla c_i(x^*) \\ &= \nabla F(x^*) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i^* \nabla c_i(x^*) = \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*) = 0\end{aligned}$$

for all penalty parameters $\theta > 0$. To show that $\nabla^2_{xx} \mathcal{L}_A(x^*, \bar{\lambda}^*; \theta) > 0$ we first calculate that

$$\nabla^2_{xx} \mathcal{L}_A(x^*, \bar{\lambda}^*; \theta) = \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) + \theta A(x^*)^\intercal A(x^*)$$

, where $A(x^*)^\intercal$ is the matrix of constraint gradients defined in the proof of theorem 4. Now suppose that the claim is wrong and that for all integers $k \geq 1$ there is a vector w_k with norm 1 such that

$$0 \geq w_k^\intercal \nabla^2_{xx} \mathcal{L}_A(x^*, \bar{\lambda}^*; k) w_k = w_k^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w_k + k ||A(x^*) w_k||^2 \tag{14}$$

. Rearranging terms gives then

$$||A(x^*) w_k||^2 \leq -(1/k) w_k^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w_k \leq -(1/k) \min_{||w||=1} w^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w \tag{15}$$

. Since the vectors w_k all lie in a compact set of $\mathbb{R}^N$ there exists a convergent subsequence $(w_{k_l})_{l \in \mathbb{N}}$ and a vector w^* with $\lim_{l \to \infty} w_{k_l} = w^*$. Inequality (15) implies that $A(x^*) w^* = 0$ and with (14) we have that

$$w_{k_l}^\intercal \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w_{k_l} \leq -k_l ||A(x^*) w_{k_l}||^2 \leq 0$$

and therefore $w^{*\intercal} \nabla^2_{xx} \mathcal{L}(x^*, \bar{\lambda}^*) w^* \leq 0$, which is a contradiction to the second order sufficient conditions. This proves the existence of a $\bar{\theta} > 0$ such that for all $\theta \geq \bar{\theta}$ we have that (13) holds. □

The theorem shows that for the exact Lagrange multiplier $\bar{\lambda}^*$ we have that x^* is also a minimizer of $\mathcal{L}_A(\cdot, \bar{\lambda}^*; \theta_k)$ for sufficiently large θ_k, which validates the approach in algorithm 2, where we have no notion of $\bar{\lambda}^*$ but try to get a good estimate $\bar{\lambda}^k$ and then perform a minimization of $\mathcal{L}_A(\cdot, \bar{\lambda}^k; \theta_k)$. By the theorem we can also hope for a good approximation of x^* even for moderate values of θ_k when the estimate $\bar{\lambda}^k$ is good. For

the next theorem we define for $x \in \mathbb{R}^N$ and $\epsilon > 0$

$$B_\epsilon(x) = \{y \in \mathbb{R}^N : ||x - y|| < \epsilon\}$$

and for a subset $X \subset \mathbb{R}^N$ and $\epsilon > 0$

$$B_\epsilon(X) = \{y \in \mathbb{R}^N : ||x - y|| < \epsilon \text{ for some } x \in X\}$$

. Additionally, we need the following lemma, which can be found in Bertsekas 1996 [p. 12]

Lemma 2 (General implicit function theorem)**.** *Let S be an open set in $\mathbb{R}^{M+N}$ and $\bar{X} \subset\subset \mathbb{R}^M$. Let $h : S \to \mathbb{R}^N$ be a function in $C^p(S)$ for some $p \geq 0$. Suppose that $\nabla_y h(x, y)$ exists and is continuous on S and that there exists a $\bar{y} \in \mathbb{R}^N$ such that $(\bar{x}, \bar{y}) \in S$, $h(\bar{x}, \bar{y}) = 0$ and $\nabla_y h(\bar{x}, \bar{y})$ is regular for all $\bar{x} \in \bar{X}$. Then there exist scalars $\delta, \epsilon > 0$ and a function $\Psi : B_\delta(\bar{X}) \to B_\epsilon(\bar{y})$ with $\Psi \in C^p(B_\delta(\bar{X}))$ such that $\Psi(\bar{x}) = \bar{y}$ for all $\bar{x} \in \bar{X}$ and $h(x, \Psi(x)) = 0$ for all $x \in B_\delta(\bar{X})$. The function is unique in the sense*

$$x \in B_\delta(\bar{X}) \wedge y \in B_\epsilon(\bar{y}) \wedge h(x, y) = 0 \Rightarrow y = \Psi(x).$$

The next theorem gives convergence properties of algorithm 2 for the more realistic case when we do not have knowledge of the exact Lagrange multiplier. The proof follows the one in Bertsekas 1996 [p. 108-111].

Theorem 6. *Suppose that the assumptions of theorem 5 hold and let $\bar{\theta}$ be the threshold parameter given by that theorem. Then there exist constants δ, ϵ and M such that the following holds*

- *For all $(\bar{\lambda}^k, \theta_k)$ in the set D defined by*

$$D = \{(\bar{\lambda}^k, \theta_k) : ||\bar{\lambda}^k - \bar{\lambda}^*|| < \delta\theta_k, \theta_k \geq \bar{\theta}\}$$

  the problem

$$\min_{x \in \mathbb{R}^N} \mathcal{L}_A(x, \bar{\lambda}^k; \theta_k) \quad \text{subject to } x \in B_\epsilon(x^*) \tag{16}$$

  has a unique solution x^k, with

$$||x^k - x^*|| \leq M||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \tag{17}$$

- *For all $(\bar{\lambda}^k, \theta_k) \in D$ it holds*

$$||\bar{\lambda}^{k+1} - \bar{\lambda}^*|| \leq M||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \tag{18}$$

  , where $\bar{\lambda}^{k+1}$ is given by the Lagrange multiplier update of algorithm 2.

- *For all $(\bar{\lambda}^k, \theta_k) \in D$, the matrix $\nabla^2_{xx}\mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)$ is positive definite and the constraint gradients $\nabla c_i(x^k)$, $i \in \mathcal{E}$, are linearly independent.*

Proof. For $\theta_k > 0$ consider the system of equations in the variables $(x, \bar{\lambda}, \bar{\lambda}^k, \theta_k)$

$$\begin{cases} \nabla F(x) - \sum_{i \in \mathcal{E}} \nabla c_i(x)\bar{\lambda}_i & = 0 \\ c_i(x) - (\bar{\lambda}_i^k - \bar{\lambda}_i)/\theta_k & = 0 \quad \forall i \in \mathcal{E} \end{cases}$$

. By defining the vector t with $t_i = (\bar{\lambda}_i^k - \bar{\lambda}_i^*)/\theta_k$, and $\gamma = 1/\theta_k$ the system can be

written as

$$\begin{cases} \nabla F(x) - \sum\limits_{i\in\mathcal{E}} \nabla c_i(x)\bar{\lambda}_i & = 0 \\ c_i(x) - t_i - \gamma\bar{\lambda}_i^* + \gamma\bar{\lambda}_i & = 0 \quad \forall i \in \mathcal{E} \end{cases} \tag{19}$$

. For $t = 0$ and $\gamma \in \left[0, 1/\bar{\theta}\right]$ system (19) has the unique solution $x = x^*$ and $\bar{\lambda} = \bar{\lambda}^*$. The Jacobian with respect to $(x, \bar{\lambda})$ evaluated at $(x^*, \bar{\lambda}^*)$ is

$$\begin{pmatrix} \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*) & -A(x^*)^\intercal \\ A(x^*) & \gamma\mathbb{I} \end{pmatrix} \tag{20}$$

, where $A(x^*)^\intercal$ is the matrix of the constraint gradients $(\nabla c_i(x^*))_{i\in\mathcal{E}}$ and $\mathbb{I} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{E}|}$ is the identity matrix. To use Lemma 2 with the compact set $\bar{X} = \{(0, \gamma) : \gamma \in \left[0, 1/\bar{\theta}\right]\}$, we need to show that this matrix is regular for all $\gamma \in \left[0, 1/\bar{\theta}\right]$. For $\gamma = 0$ this follows from the regularity of the KKT matrix when the SOSCs are fulfilled (see e.g Nocedal et Wright 2006 [p. 451-452]). For $\gamma > 0$ suppose that there are $z \in \mathbb{R}^N$ and $w \in \mathbb{R}^{|\mathcal{E}|}$, with

$$\begin{pmatrix} \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*) & -A(x^*)^\intercal \\ A(x^*) & \gamma\mathbb{I} \end{pmatrix} \begin{pmatrix} z \\ w \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

. This gives the equations

$$\begin{aligned} \nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*)z - A(x^*)^\intercal w &= 0 \\ A(x^*)z + \gamma w &= 0 \end{aligned} \tag{21}$$

. By rearranging terms in the second equation of (21) and substituting w in the first one we get

$$\left(\nabla^2_{xx}\mathcal{L}(x^*, \bar{\lambda}^*) + 1/\gamma A(x*)^\intercal A(x*)\right) z = 0$$

. Now with $\gamma = 1/\theta_k$ where $\theta_k \geq \bar{\theta}$ this is nothing but $\nabla^2_{xx}\mathcal{L}_A(x^*, \bar{\lambda}^*; \theta_k)z = 0$. From the last theorem we know that $\nabla^2_{xx}\mathcal{L}_A(x^*, \bar{\lambda}^*; \theta_k)$ is positive definite for $\theta_k \geq \bar{\theta}$ and therefore regular. This implies $z = 0$ and again with the second equation of (21) we also get $w = 0$ and therefore that the matrix in (20) is invertible. Lemma 2 gives us the existence of δ,$\epsilon > 0$ and unique continuously differentiable functions $x(t, \gamma)$ and $\bar{\lambda}(t, \gamma)$ defined on

$B_\delta(\bar{X})$ and mapping to $B_\epsilon((x^*, \bar{\lambda}^*))$ that fulfill

$$\begin{cases} \nabla F(x(t,\gamma)) - \sum\limits_{i\in\mathcal{E}} \nabla c_i(x(t,\gamma))\bar{\lambda}_i(t,\gamma) & = 0 \\ c_i(x(t,\gamma)) - t_i - \gamma\bar{\lambda}_i^* + \gamma\bar{\lambda}_i(t,\gamma) & = 0 \quad \forall i \in \mathcal{E} \end{cases} \tag{22}$$

for all $(t,\gamma) \in B_\delta(\bar{X})$. By continuity δ and ϵ can additionally be chosen such that $A(x(t,\gamma))^\intercal$ has full column rank and $\mathcal{L}_A(x(t,\gamma), \bar{\lambda}(t,\gamma); \theta_k)$ is positive definite for all $(t,\gamma) \in B_\delta(\bar{X})$ and $\theta_k \geq \bar{\theta}$. With the definitions

$$\begin{aligned} x^k &= x\left(\frac{\bar{\lambda}^k - \bar{\lambda}^*}{\theta_k}, \frac{1}{\theta_k}\right) \\ \bar{\lambda}^{k+1} &= \bar{\lambda}\left(\frac{\bar{\lambda}^k - \bar{\lambda}^*}{\theta_k}, \frac{1}{\theta_k}\right) \end{aligned} \tag{23}$$

for $\theta_k \geq \bar{\theta}$ and $||\bar{\lambda}^k - \bar{\lambda}^*|| < \delta\theta_k$ we obtain

$$\nabla F(x^k) - \sum_{i\in\mathcal{E}} \nabla c_i(x^k)\bar{\lambda}^{k+1} = 0$$

$$\bar{\lambda}_i^{k+1} = \bar{\lambda}_i^k - \theta_k c_i(x^k) \quad \forall i \in \mathcal{E}$$

and $\nabla^2_{xx}\mathcal{L}_A(x^k, \bar{\lambda}^{k+1}; \theta_k) > 0$ for all $(\bar{\lambda}^k, \theta_k) \in D$, which means that x^k fulfills the second order sufficient conditions for unconstrained optimization of problem (16) and is feasible by construction. Since the third point of this theorem also holds by construction, the only thing that is left to show are the inequalities (17) and (18). To do so, first we calculate the derivatives of (22) with respect to t and γ and get for the first equation

$$\nabla^2 F(x(t,\gamma))\nabla_t x(t,\gamma) - \sum_{i\in\mathcal{E}} \nabla^2 c_i(x(t,\gamma))\nabla_t x(t,\gamma)\bar{\lambda}_i(t,\gamma) + \nabla c_i(x(t,\gamma))\nabla_t\bar{\lambda}_i(t,\gamma)^\intercal =$$

$$\nabla^2_{xx}\mathcal{L}(x(t,\gamma), \bar{\lambda}(t,\gamma))\nabla_t x(t,\gamma) - A(x(t,\gamma))^\intercal\nabla_t\bar{\lambda}(t,\gamma) = 0$$

and

$$\nabla^2 F(x(t,\gamma))\nabla_\gamma x(t,\gamma) - \sum_{i\in\mathcal{E}} \nabla^2 c_i(x(t,\gamma))\nabla_\gamma x(t,\gamma)\bar{\lambda}_i(t,\gamma) + \nabla c_i(x(t,\gamma))\nabla_\gamma\bar{\lambda}_i(t,\gamma) =$$

$$\nabla^2_{xx}\mathcal{L}(x(t,\gamma), \bar{\lambda}(t,\gamma))\nabla_\gamma x(t,\gamma) - A(x(t,\gamma))^\intercal\nabla_\gamma\bar{\lambda}(t,\gamma) = 0$$

. If we write the remaining equations of (22) as

$$C(x(t,\gamma)) - t - \gamma\bar{\lambda}^* + \gamma\bar{\lambda}(t,\gamma) = 0$$

, where $C(x) = \big(c_1(x), c_2(x), \ldots, c_{|\mathcal{E}|}(x)\big)^\intercal$, we get

$$\begin{aligned}\nabla C(x(t,\gamma))\nabla_t x(t,\gamma) - \mathbb{I} + \gamma\nabla_t\bar{\lambda}(t,\gamma) = \\ A(x(t,\gamma))\nabla_t x(t,\gamma) - \mathbb{I} + \gamma\nabla_t\bar{\lambda}(t,\gamma) = 0\end{aligned}$$

and

$$\begin{aligned}\nabla C(x(t,\gamma))\nabla_\gamma x(t,\gamma) - \bar{\lambda}^* + \bar{\lambda}(t,\gamma) + \gamma\nabla_\gamma\bar{\lambda}(t,\gamma) = \\ A(x(t,\gamma))\nabla_\gamma x(t,\gamma) - \bar{\lambda}^* + \bar{\lambda}(t,\gamma) + \gamma\nabla_\gamma\bar{\lambda}(t,\gamma) = 0\end{aligned}$$

. Collecting all these terms results in

$$\begin{pmatrix}\nabla^2_{xx}\mathcal{L}(x(t,\gamma),\bar{\lambda}(t,\gamma)) & -A(x(t,\gamma))^\intercal \\ A(x(t,\gamma)) & \gamma\mathbb{I}\end{pmatrix}\begin{pmatrix}\nabla_t x(t,\gamma) & \nabla_\gamma x(t,\gamma) \\ \nabla_t\bar{\lambda}(t,\gamma) & \nabla_\gamma\bar{\lambda}(t,\gamma)\end{pmatrix} = \begin{pmatrix}0 & 0 \\ \mathbb{I} & \bar{\lambda}^* - \bar{\lambda}(t,\gamma)\end{pmatrix}$$

and we get that for all (t,γ) with $||t|| < \delta$ and $\gamma \in \big[0, 1/\overline{\theta}\big]$

$$\nabla\begin{pmatrix}x(t,\gamma) \\ \bar{\lambda}(t,\gamma)\end{pmatrix} = K(t,\gamma)\begin{pmatrix}0 & 0 \\ \mathbb{I} & \bar{\lambda}^* - \bar{\lambda}(t,\gamma)\end{pmatrix}$$

, where $K(t,\gamma)$ is defined as

$$K(t,\gamma) = \begin{pmatrix}\nabla^2_{xx}\mathcal{L}(x(t,\gamma),\bar{\lambda}(t,\gamma)) & -A(x(t,\gamma))^\intercal \\ A(x(t,\gamma)) & \gamma\mathbb{I}\end{pmatrix}^{-1}$$

. Since the matrix in (20) is regular for $\gamma \in \big[0, 1/\overline{\theta}\big]$, we have that $K(t,\gamma)$ is uniformly bounded on $\{(t,\gamma) : ||t|| < \delta, \gamma \in \big[0, 1/\overline{\theta}\big]\}$ for sufficiently small δ. Let κ be such that $||K(t,\gamma)|| \leq \kappa$ for all $||t|| < \delta$ and $\gamma \in \big[0, 1/\overline{\theta}\big]$ and if necessary choose δ additionally small enough such that $\kappa\delta < 1$. Due to

$$\begin{pmatrix}x(t,\gamma) - x^* \\ \bar{\lambda}(t,\gamma) - \bar{\lambda}^*\end{pmatrix} = \begin{pmatrix}x(t,\gamma) - x(0,0) \\ \bar{\lambda}(t,\gamma) - \bar{\lambda}(0,0)\end{pmatrix} = \int_0^1 K(\xi t, \xi\gamma)\begin{pmatrix}0 & 0 \\ \mathbb{I} & \bar{\lambda}^* - \bar{\lambda}(\xi t, \xi\gamma)\end{pmatrix}\begin{pmatrix}t \\ \gamma\end{pmatrix} d\xi$$

we then have

$$\big(||x(t,\gamma) - x^*||^2 + ||\bar{\lambda}(t,\gamma) - \bar{\lambda}^*||^2\big)^{\frac{1}{2}} \leq \kappa\left(||t|| + \gamma \max_{0\leq\xi\leq 1} ||\bar{\lambda}(\xi t, \xi\gamma) - \bar{\lambda}^*||\right)$$

, which implies

$$||\bar{\lambda}(t,\gamma) - \bar{\lambda}^*|| \leq \kappa||t|| + \kappa\gamma \max_{0\leq\xi\leq 1} ||\bar{\lambda}(\xi t, \xi\gamma) - \bar{\lambda}^*||$$

for all $||t|| < \delta$, $\gamma \in \left[0, 1/\bar{\theta}\right]$ and $\gamma < \delta$. Substituting ξt and $\xi\gamma$ with $\xi \in [0,1]$ for t and γ in this inequality results in

$$\max_{0\leq\xi\leq 1} ||\bar{\lambda}(\xi t, \xi\gamma) - \bar{\lambda}^*|| \leq \frac{\kappa}{1-\kappa\gamma}||t||$$

. Note that $\kappa\gamma < \kappa\delta < 1$. Altogether, we have that

$$\left(||x(t,\gamma) - x^*||^2 + ||\bar{\lambda}(t,\gamma) - \bar{\lambda}^*||^2\right)^{\frac{1}{2}} \leq \left(\kappa + \frac{\kappa^2\gamma}{1-\kappa\gamma}\right)||t|| \leq \frac{\kappa}{1-\kappa\delta}||t||$$

and by taking δ again sufficiently small such that $\kappa/\left(1-\kappa\delta\right) \leq 2\kappa$

$$\left(||x(t,\gamma) - x^*||^2 + ||\bar{\lambda}(t,\gamma) - \bar{\lambda}^*||^2\right)^{\frac{1}{2}} \leq M_1||t||$$

, where $M_1 = 2\kappa$. Using the definitions (23) again we have that for all $(\bar{\lambda}^k, \theta_k)$ with $||\bar{\lambda}^k - \bar{\lambda}^*|| < \delta\theta_k$ and $\theta_k > \max\{\bar{\theta}, 1/\delta\}$

$$\begin{aligned} ||x^k - x^*|| &\leq M_1||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \\ ||\bar{\lambda}^{k+1} - \bar{\lambda}^*|| &\leq M_1||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \end{aligned}$$

. Since x and $\bar{\lambda}$ are continuously differentiable, we can also find an $M_2 > 0$ such that for all $(\bar{\lambda}^k, \theta_k)$ with $||\bar{\lambda}^k - \bar{\lambda}^*|| < \delta\theta_k$ and $\bar{\theta} \leq \theta_k \leq \max\{\bar{\theta}, 1/\delta\}$ it holds

$$\begin{aligned} ||x^k - x^*|| &\leq M_2||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \\ ||\bar{\lambda}^{k+1} - \bar{\lambda}^*|| &\leq M_2||\bar{\lambda}^k - \bar{\lambda}^*||/\theta_k \end{aligned}$$

. Equation (17) and (18) therefore hold for all $(\bar{\lambda}^k, \theta_k)$ with $||\bar{\lambda}^k - \bar{\lambda}^*|| < \delta\theta_k$ and $\theta_k \geq \bar{\theta}$ with $M = \max\{M_1, M_2\}$. □

The theorem shows that x^k will be close to x^*, if the estimate $\bar{\lambda}^k$ is accurate or the penalty parameter θ_k is large, which means that we have two possibilities to increase the accuracy of x^k, whereas in the quadratic penalty method we can only increase the penalty parameter. It also shows that we can locally improve the accuracy of the Lagrange multiplier estimate by choosing a sufficiently larger penalty parameter. The last point shows that the second order sufficient conditions for unconstrained minimization

are fulfilled for the k-th subproblem, which allows to hope for a good performance of unconstrained minimization techniques.

3.3 Bound-constrained Lagrangian method

With regard to the model, this chapter finally concerns problems of the form

$$\min_{x\in\mathbb{R}^N} F(x) \quad \text{subject to } \begin{cases} c_i(x) = 0 & \forall i \in \mathcal{E} \\ l \leq x \leq u \end{cases} \tag{24}$$

with twice continuously differentiable objective function F and constraint functions c_i and vectors l and u in $(\mathbb{R} \cup \{-\infty, \infty\})^N$, with $l_i < u_i$ for all $i \in \{1, \ldots, N\}$. The bound-constraints have to be understood componentwise and infinite entries of l and u correspond to unbounded variables in the respective direction. The bound-constrained Lagrangian method enforces the bound-constraints in the subproblem and incorporates only the equality constraints into the augmented Lagrangian. The k-th subproblem therefore has the form

$$\min_{x\in\mathbb{R}^N} \mathcal{L}_A(x, \bar{\lambda}^k; \theta_k) \quad \text{subject to } \; l \leq x \leq u \tag{25}$$

. An $x^* \in \mathbb{R}^N$ is a KKT point of problem (25) if and only if

$$x^* - P(x^* - \nabla_x \mathcal{L}_A(x^*, \bar{\lambda}^k; \theta_k), l, u) = 0 \tag{26}$$

, where $P(g, l, u)$ for a vector $g \in \mathbb{R}^N$ is the projection onto the box defined by l and u, i.e.

$$P(g, l, u)_i = \begin{cases} l_i & \text{if } g_i \leq l_i \\ g_i & \text{if } g_i \in (l_i, u_i) \\ u_i & \text{if } g_i \geq u_i \end{cases}$$

, like the following theorem shows. It can be found as exercise 17.9 in Nocedal et Wright 2006 [p. 528].

Theorem 7. *An $x^* \in \mathbb{R}^N$ is a KKT point of the problem*

$$\min_{x \in \mathbb{R}^N} \phi(x) \text{ subject to } \; l \leq x \leq u$$

with $\phi \in C^1$ and vectors l and u with $l < u$ iff

$$x^* - P(x^* - \nabla\phi(x^*), l, u) = 0$$

Proof. The Lagrangian of the given problem can be written in the form

$$\mathcal{L}(x, \bar{\rho}^1, \bar{\rho}^2) = \phi(x) - \bar{\rho}^{1\intercal}(x - l) - \bar{\rho}^{2\intercal}(u - x)$$

. The gradient with respect to x is

$$\nabla_x \mathcal{L}(x, \bar{\rho}^1, \bar{\rho}^2) = \nabla\phi(x) - \bar{\rho}^1 + \bar{\rho}^2$$

and the KKT conditions can therefore be formulated as

$$\begin{aligned}
\nabla\phi(x) &= \bar{\rho}^1 - \bar{\rho}^2 \\
\bar{\rho}_i^1(x_i - l_i) &= 0 \quad \forall i \in \{1, \ldots, N\} \\
\bar{\rho}_i^2(u_i - x_i) &= 0 \quad \forall i \in \{1, \ldots, N\} \\
\bar{\rho}^1 &\geq 0 \\
\bar{\rho}^2 &\geq 0 \text{ and } \; l \leq x \leq u
\end{aligned}$$

. Suppose now that x^* fulfills all this conditions. We then have $\nabla\phi(x^*) = \bar{\rho}^1 - \bar{\rho}^2$ and due to the complementary conditions we have that $l_i < x_i^* < u_i$ implies that $\bar{\rho}_i^1 = \bar{\rho}_i^2 = 0$. Additionally, $x_i^* = l_i$ implies that $x_i^* < u_i$ and therefore that $\bar{\rho}_i^2 = 0$. In the same way $x_i^* = u_i$ implies that $\bar{\rho}_i^1 = 0$. Collecting all these conditions we get

$$\nabla\phi(x^*)_i = \begin{cases} 0 & \text{if } l_i < x_i^* < u_i \\ \bar{\rho}_i^1 & \text{if } x_i^* = l_i \\ -\bar{\rho}_i^2 & \text{if } x_i^* = u_i \end{cases}$$

, which subsequently leads to

$$(x^* - \nabla\phi(x^*))_i = \begin{cases} x_i^* & \text{if } l_i < x_i^* < u_i \\ x_i^* - \bar{\rho}_i^1 & \text{if } x_i^* = l_i \\ x_i^* + \bar{\rho}_i^2 & \text{if } x_i^* = u_i \end{cases}$$

. Since $\bar{\rho}^1, \bar{\rho}^2 \geq 0$ this results in $P(x^* - \nabla\phi(x^*), l, u) = x^*$. For the other direction suppose now, that we have an $x^* \in \mathbb{R}^N$ such that $x^* - P(x^* - \nabla\phi(x^*), l, u) = 0$ holds. Since $x^* = P(x^* - \nabla\phi(x^*), l, u)$, the vector x^* is feasible. Additionally, we can write this equation in the following form

$$\begin{aligned} x^* &= P(x^* - \nabla\phi(x^*), l, u) \\ &= x^* - \nabla\phi(x^*) + \sum_{i \in I_1}(l_i - (x_i^* - \nabla\phi(x^*)_i))e_i - \sum_{i \in I_2}((x_i^* - \nabla\phi(x^*)_i) - u_i)e_i \end{aligned} \tag{27}$$

, where $I_1 = \{i : (x^* - \nabla\phi(x^*))_i \leq l_i\}$, $I_2 = \{i : (x^* - \nabla\phi(x^*))_i \geq u_i\}$ and the vector e_i is the i-th canonical basis vector of $\mathbb{R}^N$. If we now define

$$\bar{\rho}_i^1 = \begin{cases} l_i - (x_i^* - \nabla\phi(x^*)_i) & \text{if } x_i^* - \nabla\phi(x^*)_i \leq l_i \\ 0 & \text{else} \end{cases}$$

$$\bar{\rho}_i^2 = \begin{cases} (x_i^* - \nabla\phi(x^*)_i) - u_i & \text{if } x_i^* - \nabla\phi(x^*)_i \geq u_i \\ 0 & \text{else} \end{cases}$$

, equation (27) gives $\nabla\phi(x^*) = \bar{\rho}^1 - \bar{\rho}^2$. By definition it also holds that $\bar{\rho}^1, \bar{\rho}^2 \geq 0$. So what is left to show is, that $\bar{\rho}_i^1(x_i - l_i) = 0$ and $\bar{\rho}_i^2(u_i - x_i) = 0$ holds for all $i \in \{1, \ldots, n\}$. We will prove the first one as the second one can be done in the same way. Let $i \in \{1, \ldots, n\}$ be arbitrary. Since x^* is feasible we can split the proof in two cases.

Case 1: $x_i^* = l_i$
In this case it follows immediately that $\bar{\rho}_i^1(x_i - l_i) = 0$ holds.
Case 2: $x_i^* > l_i$
We split this case again in two cases
Case 2.1: $x_i^* - \phi(x^*)_i \leq l_i$
Since $x_i^* - \phi(x^*)_i \leq l_i < u_i$, we have by definition of $\bar{\rho}^2$ that $\bar{\rho}_i^2 = 0$. Now $\nabla\phi(x^*)_i = \bar{\rho}_i^1 - \bar{\rho}_i^2 = \bar{\rho}_i^1$, and therefore we have again by definition of $\bar{\rho}^1$ that $\bar{\rho}_i^1 = l_i - x_i^* + \nabla\phi(x^*)_i = l_i - x_i^* + \bar{\rho}_i^1$, which gives $x_i^* = l_i$ and contradicts the assumption $x_i^* > l_i$.

Case 2.2: $x_i^* - \phi(x^*)_i > l_i$
This implies by definition of $\bar{\rho}^1$ that $\bar{\rho}_i^1 = 0$ and therefore $\bar{\rho}_i^1(x_i - l_i) = 0$. □

The theorem shows, that we can use equation (26) as a stopping criteria for the minimization problem (25). Below, we gather the constraint functions c_i again in a function $C = (c_1, c_2, \ldots, c_{|\mathcal{E}|})^\intercal$. A general framework for the bound-constrained Lagrangian method can then be specified as (see Nocedal et Wright 2006 [p.520-521]):

Data: starting point x_s^0, starting Lagrange multiplier $\bar{\lambda}^0$, convergence tolerances η_* and ω_*
Result: approximate solution x^*, approximate Lagrange multiplier $\bar{\lambda}^*$
Set $\theta_0 = 10$, $\omega_0 = 1/\theta_0$ and $\eta_0 = 1/\theta_0^{0.1}$;
for $k = 0, 1, 2, \ldots$ **do**
  Find approximate minimizer x^k of $\min_{x \in \mathbb{R}^N} \mathcal{L}_A(x, \bar{\lambda}^k; \theta_k)$ subject to $l \leq x \leq u$ starting at x_s^k with $||x^k - P(x^k - \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k), l, u)|| \leq \omega_k$;
  if $||C(x_k)|| \leq \eta_k$ **then**
    if $||C(x_k)|| \leq \eta_*$ *and* $||x^k - P(x^k - \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k), l, u)|| \leq \omega_*$ **then**
      $\bar{\lambda}^{k+1} = \bar{\lambda}^k - \theta_k C(x^k)$;
      return $x^* = x^k$ and $\bar{\lambda}^* = \bar{\lambda}^{k+1}$;
    end
    $\bar{\lambda}^{k+1} = \bar{\lambda}^k - \theta_k C(x^k)$;
    $\theta_{k+1} = \theta_k$;
    $\eta_{k+1} = \eta_k / \theta_{k+1}^{0.9}$;
    $\omega_{k+1} = \omega_k / \theta_{k+1}$;
  else
    $\bar{\lambda}^{k+1} = \bar{\lambda}^k$;
    $\theta_{k+1} = 100 \cdot \theta_k$;
    $\eta_{k+1} = 1/\theta_{k+1}^{0.1}$;
    $\omega_{k+1} = 1/\theta_{k+1}$;
  end
  $x_s^{k+1} = x^k$;
end

Algorithm 3: Bound-constrained Lagrangian Method

. The main work in algorithm 3 takes place in $\min_{x \in \mathbb{R}^N} \mathcal{L}_A(x, \bar{\lambda}^k; \theta_k)$ subject to $l \leq x \leq u$. How such a minimization is done will be shown in the next chapter. After that the condition $||C(x_k)|| \leq \eta_k$ measures if the constraints have decreased sufficiently. If the

condition holds, we do not change the penalty parameter as it produces an acceptable value of constraint violation. We update the Lagrange multiplier estimate and tighten the tolerances η_k and ω_k for the next iteration. If it does not hold, we increase the penalty parameter and do not update the Lagrange multiplier estimate, as the next step should focus on improving feasibility. The next section gives conditions under which algorithm 3 produces a KKT point of problem (24).

3.3.1 Convergence Analysis

The theory that gets presented in this chapter follows Conn et al. 1991. To begin with, we introduce the concept of dominated and floating variables. To do so, we first define the compact notation

$$P(x, v) = x - P(x - v, l, u)$$

, where $P(x - v, l, u)$ is the projection onto the feasible box of the vector $x - v$, defined in chapter 3.3. Furthermore, over the whole chapter we denote with $\mathcal{L}^P(x, \bar{\lambda}, \bar{\rho}^1, \bar{\rho}^2)$ the Lagrangian of problem (24), i.e.

$$\mathcal{L}^P(x, \bar{\lambda}, \bar{\rho}_1, \bar{\rho}_2) = F(x) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i c_i(x) - \bar{\rho}^1(x - l) - \bar{\rho}^2(u - x)$$

, whereas $\mathcal{L}(x, \bar{\lambda})$ denotes the Lagrangian of problem (24) where the bound constraints are ignored, i.e.

$$\mathcal{L}(x, \bar{\lambda}) = F(x) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i c_i(x)$$

and $\mathcal{L}_A(x, \bar{\lambda}, \theta)$ denotes the augmented Lagrangian defined in chapter 3.2.2. Now, for any sequence of vectors $l \leq x^k \leq u$ in the framework of algorithm 3, we have three mutually exclusive possibilities for each component x_i^k

$$\begin{array}{ll} (i) & 0 \leq x_i^k - l_i \leq \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i \\ (ii) & \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i \leq x_i^k - u_i \leq 0 \\ (iii) & x_i^k - u_i < \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i < x_i^k - l_i \end{array}$$

We call variables x_i^k that fulfill (i) dominated from above, variables that fulfill (ii) dominated from below and variables that fulfill (iii) floating. Variables that fulfill (i) or (ii)

are called dominated. In case (i) we have

$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = x_i^k - l_i$$

, whereas in case (ii) we have

$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = x_i^k - u_i$$

and in case (iii) we have

$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i$$

. Algorithm 3 forces $P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))$ to zero, thus we have that the dominated variables are pushed to their bounds while the floating variables can find their own level. Now, for a convergent sequence $(x^k)_{k\in\mathbb{N}}$ with $x^k \to x^*$, we define the index sets

$$\begin{aligned}
I_1 &= \{i : x_i^k \text{ are floating for sufficiently large k and } l_i < x_i^* < u_i\} \\
I_{2l} &= \{i : x_i^k \text{ are dominated from above for sufficiently large k}\} \\
I_{2u} &= \{i : x_i^k \text{ are dominated from below for sufficiently large k}\} \\
I_{3l} &= \{i : x_i^k \text{ are floating for sufficiently large k and } x_i^* = l_i\} \\
I_{3u} &= \{i : x_i^k \text{ are floating for sufficiently large k and } x_i^* = u_i\} \\
I_2 &= I_{2l} \cup I_{2u} \\
I_3 &= I_{3l} \cup I_{3u} \\
I_4 &= (I_1 \cup I_2 \cup I_3)^C
\end{aligned}$$

. If the vectors x^k are now chosen such that $P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))$ goes to zero for $k \to \infty$, we can state the first theorem.

Theorem 8. *Suppose that $(x^k)_{k\in\mathbb{N}}$ is a convergent sequence with limit point x^* and that $P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))$ goes to zero for $k \to \infty$. Then*

(i) The variables in the set I_2,I_3 and I_4 all converge to their bounds.

(ii) The components $\nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i$ with $i \in I_1$ or $i \in I_3$ converge to zero

(iii) If a component $\nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i$ with $i \in I_4$ converges to a finite limit, the limit is zero

Proof. Since for indices in I_2 we have

$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = x_i^k - l_i \quad \forall i \in I_{2l}$$
$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = x_i^k - u_i \quad \forall i \in I_{2u}$$

point (i) holds true. For indices in I_3 point (i) holds by definition and for indices in I_4 we can find a subsequence x^{k_l} such that $P(x^{k_l}, \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^{k_l}; \theta_{k_l}))_i = x_i^{k_l} - l_i$ or $P(x^{k_l}, \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^{k_l}; \theta_{k_l}))_i = x_i^{k_l} - u_i$ holds and therefore point (i) also holds for $i \in I_4$. Point (ii) holds true since for $i \in I_1$ or $i \in I_3$, we have that

$$P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))_i = \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)_i$$

. Point (iii) holds true since we can again find a subsequence x^{k_l} such that

$$P(x^{k_l}, \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^{k_l}; \theta_{k_l}))_i = \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^{k_l}; \theta_{k_l})_i$$

holds for an $i \in I_4$ with the given assumptions. □

For the next part we denote with $\nabla \hat{F}(x)$ the components of $\nabla F(x)$, which correspond to the index set I_1 and with $\hat{A}(x)^\intercal$ the submatrix of the matrix of constrained gradients, defined in theorem 4, which correspond to the columns of $\hat{A}(x)^\intercal$ with index in I_1. Moreover, we denote with $\hat{M}$ the matrix formed by the columns of any matrix M with indices in I_1. With regard to theorem 4 we define the Lagrange multiplier estimate

$$\bar{\lambda}(x) = (\hat{A}(x)^+)^\intercal \nabla \hat{F}(x)$$

for all x, where the right generalized inverse

$$\hat{A}(x)^+ = \hat{A}(x)^\intercal \left(\hat{A}(x) \hat{A}(x)^\intercal \right)^{-1}$$

is well defined. The following lemma is given without proof.

Lemma 3. *If $\hat{A}(x)\hat{A}(x)^\intercal$ is regular the Lagrange multiplier estimate $\bar{\lambda}(x)$ is differentiable and it holds*

$$\nabla\bar{\lambda}(x) = (\hat{A}(x)^+)^\intercal\nabla^2_{xx}\hat{\mathcal{L}}(x,\bar{\lambda}(x)) + \left(\hat{A}(x)\hat{A}(x)^\intercal\right)^{-1}R(x)$$

, where the i-th row of $R(x)$ is defined as $\left(\nabla\hat{F}(x) - \hat{A}(x)^\intercal\bar{\lambda}(x)\right)^\intercal\nabla^2 c_i(x)$.

Now we are ready to give the first general convergence result.

Theorem 9. *Let $l \leq x^k \leq u$ be a convergent sequence of vectors that lie within a closed bounded domain of $\mathbb{R}^N$, with limit point x^* at which the LICQ holds. Let $\bar{\lambda}^* = \bar{\lambda}(x^*)$ and let $\bar{\lambda}^k$ be the sequence of Lagrange multiplier estimates given by algorithm 3. Additionally, suppose that $(\theta_k)_{k\in\mathbb{N}}$ form a non decreasing sequence of positive scalars and that $(\omega_k)_{k\in\mathbb{N}}$ forms a sequence of positive scalars that converge to zero. Furthermore, let*

$$||P(x^k, \nabla_x\mathcal{L}_A(x^k,\bar{\lambda}^k;\theta_k))|| \leq \omega_k$$

. Then there exist positive scalars a_1 and a_2 and an integer k_0 such that

$$||\bar{\lambda}^{k+1} - \bar{\lambda}^*|| \leq a_1\omega_k + a_2||x^k - x^*|| \tag{28}$$

, where $\bar{\lambda}^{k+1}$ is given by the Lagrange multiplier update of algorithm 3.

$$||\bar{\lambda}(x^k) - \bar{\lambda}^*|| \leq a_2||x^k - x^*|| \tag{29}$$

and

$$||C(x^k)|| \leq \frac{1}{\theta_k}\left(||\bar{\lambda}^k - \bar{\lambda}^*|| + a_1\omega_k + a_2||x^k - x^*||\right) \tag{30}$$

for all $k \geq k_0$.
If in addition $C(x^*) = 0$, then x^* is a KKT point of problem (24) and $\bar{\lambda}^*$ is the corresponding Lagrange multiplier for the equality constraints. Furthermore, the sequences $\bar{\lambda}^k$ and $\bar{\lambda}(x^k)$ converge to $\bar{\lambda}^*$.

Proof. Since all the x^k lie within a closed bounded domain and converge to x^*, at which the LICQ holds, we have that for k sufficiently large $\hat{A}(x^k)^+$ exists, is bounded and converges to $\hat{A}(x^*)$. Therefore we have a constant a_1 such that

$$||(\hat{A}(x^k)^+)^\intercal|| \leq a_1$$

for sufficiently large k. By differentiating $\mathcal{L}_A(x, \bar{\lambda}; \theta)$ with respect to x, i.e.

$$\begin{aligned}\nabla_x \mathcal{L}_A(x, \bar{\lambda}; \theta) &= \nabla F(x) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i \nabla c_i(x) + \theta \sum_{i \in \mathcal{E}} c_i(x) \nabla c_i(x) \\ &= \nabla F(x) - \sum_{i \in \mathcal{E}} (\bar{\lambda}_i - \theta c_i(x)) \nabla c_i(x)\end{aligned}$$

, one gets $\nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k) = \nabla_x \mathcal{L}(x^k, \bar{\lambda}^{k+1})$. Since $||P(x^k, \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k))|| \leq \omega_k$ and the variables with indices in the set I_1 are floating we have

$$||\nabla \hat{F}(x) - \hat{A}(x^k)^\intercal \bar{\lambda}^{k+1}|| \leq \omega_k$$

. Thus we have

$$\begin{aligned}||\bar{\lambda}^{k+1} - \bar{\lambda}(x^k)|| &= ||\bar{\lambda}^{k+1} - (\hat{A}(x^k)^+)^\intercal \nabla \hat{F}(x^k)|| \\ &= ||(\hat{A}(x^k)^+)^\intercal \left(\hat{A}(x^k)^\intercal \bar{\lambda}^{k+1} - \nabla \hat{F}(x^k) \right)|| \\ &\leq ||(\hat{A}(x^k)^+)^\intercal|| \omega_k \leq a_1 \omega_k\end{aligned}$$

. By Lemma 3 we have that $\bar{\lambda}(x)$ is differntiable in a neighbourhood of x^* and therefore we get

$$\bar{\lambda}(x^k) - \bar{\lambda}(x^*) = \int_0^1 \nabla \bar{\lambda}(x^k + \delta(x^* - x^k)) \cdot (x^k - x^*) d\delta$$

. Since the term $\nabla \bar{\lambda}(x)$ is bounded for x sufficiently near to x^* we have that

$$||\bar{\lambda}(x^k) - \bar{\lambda}(x^*)|| \leq a_2 ||x^k - x^*||$$

for some $a_2 > 0$, which gives us (29). We also obtain (28) by

$$\begin{aligned}||\bar{\lambda}^{k+1} - \bar{\lambda}^*|| &\leq ||\bar{\lambda}^{k+1} - \bar{\lambda}(x^k)|| + ||\bar{\lambda}(x^k) - \bar{\lambda}^*|| \\ &\leq a_1 \omega_k + a_2 ||x^k - x^*||\end{aligned}$$

. By rearranging terms in the Lagrange multiplier update we have

$$\begin{aligned} C(x^k) &= \frac{1}{\theta_k}\left(\bar{\lambda}^k - \bar{\lambda}^{k+1}\right) \\ &= \frac{1}{\theta_k}\left(\left(\bar{\lambda}^k - \bar{\lambda}^*\right) + \left(\bar{\lambda}^* - \bar{\lambda}^{k+1}\right)\right) \end{aligned}$$

and therefore we get (30) by

$$\begin{aligned} ||C(x^k)|| &\leq \frac{1}{\theta_k}\left(||\bar{\lambda}^k - \bar{\lambda}^*|| + ||\bar{\lambda}^* - \bar{\lambda}^{k+1}||\right) \\ &\leq \frac{1}{\theta_k}\left(||\bar{\lambda}^k - \bar{\lambda}^*|| + a_1\omega_k + a_2||x^k - x^*||\right) \end{aligned}$$

. Now, suppose that $C(x^*) = 0$. Since $\omega_k \to 0$ we have by (28) that $\bar{\lambda}^k \to \bar{\lambda}^*$. Furthermore, we have $\nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k) \to \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)$. Theorem 8 then gives us

$$\begin{array}{rcll} l_i < x_i^* < u_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i = 0 & \forall i \in I_1 \\ x_i^* = l_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i \geq 0 & \forall i \in I_{2l} \\ x_i^* = u_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i \leq 0 & \forall i \in I_{2u} \\ x_i^* = l_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i = 0 & \forall i \in I_{3l} \\ x_i^* = u_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i = 0 & \forall i \in I_{3u} \\ x_i^* = l_i \text{ or } x_i^* = u_i & \text{and} & \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*)_i = 0 & \forall i \in I_4 \end{array} \tag{31}$$

. If we now define the two vectors

$$\bar{\rho}^1 = \begin{cases} \nabla \mathcal{L}(x^*, \bar{\lambda}^*)_i & i \in I_{2l} \\ 0 & else \end{cases} \qquad \bar{\rho}^2 = \begin{cases} -\nabla \mathcal{L}(x^*, \bar{\lambda}^*)_i & i \in I_{2u} \\ 0 & else \end{cases}$$

, we have

$$\begin{aligned} \nabla_x \mathcal{L}^P(x^*, \bar{\lambda}^*, \bar{\rho}^1, \bar{\rho}^2) &= \nabla F(x) - \sum_{i \in \mathcal{E}} \bar{\lambda}_i \nabla c_i(x) - \bar{\rho}^1 + \bar{\rho}^2 \\ &= \nabla_x \mathcal{L}(x^*, \bar{\lambda}^*) - \bar{\rho}^1 + \bar{\rho}^2 = 0 \end{aligned}$$

. Furthermore, $\bar{\rho}^1, \bar{\rho}^2 \geq 0$ and the complementary condition holds by (31). Thus, we have that x^* is a KKT point of problem (24). The convergence of $\bar{\lambda}(x^k)$ follows by (29). □

For the main result we need an additional theorem, which shows that the Lagrange multiplier estimates of algorithm 3 do not behave too badly.

Theorem 10. *Suppose that the penalty parameters θ_k converge to infinity while algorithm 3 is executed, then we have that*

$$\frac{1}{\theta_k}||\bar{\lambda}^k|| \to 0 \text{ for } k \to \infty$$

Proof. Suppose that $\theta_k \to \infty$ in algorithm 3, then we have that the condition $||C(x^k)|| \leq \eta_k$ is violated infinitely often. Let $K = \{k_0, k_1, k_2, \dots\}$ be the set of indices in which this condition is violated and it holds

$$\theta_{k_i} \geq 2^{\frac{1}{0.9}} \quad \forall k_i \in K \tag{32}$$

. We look at the Lagrange multiplier updates between two successive indices in K. Therefore, let $k_i + j$ be such that $k_i < k_i + j \leq k_{i+1}$, we then have

$$\bar{\lambda}^{k_i+j} = \bar{\lambda}^{k_i} - \sum_{l=1}^{j-1} \theta_{k_i+l} C(x^{k_i+l}) \tag{33}$$

, with the convention that empty sums are zero. Furthermore, we have that no penalty parameter update takes place between two successive indices in K, and therefore

$$\theta_{k_i+j} = \theta_{k_i+1} = 100 \cdot \theta_{k_i} \tag{34}$$

. Suppose that $j > 1$ now. We have look at the iterations $k_i + l$ with $1 \leq l < j$. For those we know that the condition $||C(x^{k_i+l})|| \leq \eta_{k_i+l}$ holds and therefore due to the recursive definition of η_k, we have that

$$||C(x^{k_i+l})|| \leq \frac{1}{\theta_{k_i+1}^{0.1+(l-1)0.9}} \tag{35}$$

. Combining (32),(33) and (35) we get

$$\begin{aligned}||\bar{\lambda}^{k_i+j}|| &\leq ||\bar{\lambda}^{k_i}|| + \sum_{l=1}^{j-1} \theta_{k_i+l}||C(x^{k_i+l})|| \leq ||\bar{\lambda}^{k_i}|| + \sum_{l=1}^{j-1} \theta_{k_i+1} \frac{1}{\theta_{k_i+1}^{0.1+(l-1)0.9}} \\ &\leq ||\bar{\lambda}^{k_i}|| + \theta_{k_i+1}^{0.9} \sum_{l=1}^{j-1} \left(\frac{1}{\theta_{k_i+1}^{0.9}}\right)^{l-1} \leq ||\bar{\lambda}^{k_i}|| + \theta_{k_i+1}^{0.9} \frac{1}{1-\frac{1}{\theta_{k_i+1}^{0.9}}} \\ &\leq ||\bar{\lambda}^{k_i}|| + 2 \cdot \theta_{k_i+1}^{0.9}\end{aligned}$$

, where we used $\theta_{k_i+l} = \theta_{k_i+1}$ for all $1 \leq l < j$. By dividing through θ_{k_i+j} and using (34) we get

$$\begin{aligned}\frac{1}{\theta_{k_i+j}}||\bar{\lambda}^{k_i+j}|| &\leq \frac{1}{100}\frac{1}{\theta_{k_i}}||\bar{\lambda}^{k_i}|| + 2\theta_{k_i+1}^{-0.1} \\ &\leq \frac{1}{100}\frac{1}{\theta_{k_i}}||\bar{\lambda}^{k_i}|| + 2\frac{1}{100^{0.1}}\theta_{k_i}^{-0.1}\end{aligned} \tag{36}$$

. This inequality is also satisfied for $j = 1$, since

$$\frac{1}{\theta_{k_i+1}}||\bar{\lambda}^{k_i+1}|| = \frac{1}{100}\frac{1}{\theta_{k_i}}||\bar{\lambda}^{k_i}||$$

. If we now define

$$\alpha_i = \tfrac{1}{\theta_{k_i}}||\bar{\lambda}^{k_i}||, \quad \beta_i = 2 \cdot \theta_{k_i}^{-0.1}, \quad \kappa = \tfrac{1}{100}$$

, we get the that

$$\begin{aligned}\alpha_{i+1} &\leq \kappa\alpha_i + \kappa^{0.1}\beta_i \\ \beta_{i+1} &= \kappa^{0.1}\beta_i\end{aligned} \tag{37}$$

, which leads by induction to

$$\begin{aligned}0 \leq \alpha_i &\leq \kappa^i\alpha_0 + \kappa^{0.1\cdot i}\sum_{l=0}^{i-1}\left(\kappa^{1-0.1}\right)^l\beta_0 \\ &\leq \kappa^i\alpha_0 + \kappa^{0.1\cdot i}\frac{\beta_0}{1-\kappa^{0.9}}\end{aligned}$$

and therefore $\alpha_i \to 0$. By (37) also $\beta_i \to 0$ and therefore the right-hand side of (36) converges to zero which ends the proof. □

Finally, we can state the main convergence result:

Theorem 11. *Let $(x_k)_{k\in\mathbb{N}}$ be a sequence of vectors that lie within a closed bounded domain of $\mathbb{R}^N$ and is generated by algorithm 3. Then, every accumulation point x^* of $(x_k)_{k\in\mathbb{N}}$ for which the LICQ holds is a KKT point of problem (24) and the sequence of Lagrange multiplier estimates converges to the corresponding Lagrange multiplier vector $\bar{\lambda}^*$ for the equality constraints.*

Proof. The result follows by theorem 9, if we can show that $C(x^*) = 0$. Let $(x^{k_l})_{l\in\mathbb{N}}$ be a subsequence of $(x^k)_{k\in\mathbb{N}}$ with $\lim_{l\to\infty} x^{k_l} = x^*$. Suppose first, that the penalty parameters θ_k stay bounded in the execution of algorithm 3. If this the case, the condition

$$||C(x^{k_l})|| \leq \eta_{k_l}$$

holds true in every iteration for sufficiently large k_l. Since $\eta_{k_l} \to 0$ we have that $C(x^*) = 0$. Now suppose to the contrary, that $\theta_k \to \infty$. Then by theorem 10 it follows that

$$\frac{1}{\theta_{k_l}}||\bar{\lambda}^{k_l} - \bar{\lambda}^*|| \to 0$$

and by inequality (30) of theorem 9 we have that $C(x^{k_l}) \to 0$, since $\omega_{k_l} \to 0$. □

Going back to the model we have that the constraint matrix C has full row rank and therefore the LICQ is fulfilled independently of x^*. Furthermore, although the vector of upper bounds u contains infinite entries, the slack variables s_k are bounded by

$$L - \epsilon \sum_{i=1}^{n} g_i$$

. Theorem 11 therefore implies that every accumulation point of algorithm 3 is a KKT point in our case. The last question that is unanswered in algorithm 3 is how to solve subproblem (25). The next chapter addresses this issue.

3.4 Gradient projection method

The gradient projection method is an active set method that is able to identify the correct active set in finitely many iterations due to a powerful choice of the working set. Every iteration proceeds in two steps. First, it identifies the so-called generalized cauchypoint, which is the first minimizer along the projected steepest decent direction. Then, it

fixes the variables which are active at the generalized cauchy point to their bounds and performs a subspace minimization on the remaining variables. We will first have a look at quadratic programs with bound constraints, to explain the two essential steps of the gradient projection method. After that, a trust region based gradient projection method for general bound-constrained problems will be presented.

3.4.1 Gradient projection method for bound-constrained QPs

In this chapter, we concentrate on the problem

$$\min_{x\in\mathbb{R}^N} F(x) = \frac{1}{2}x^\intercal Gx + x^\intercal c \quad \text{subject to } l \leq x \leq u \tag{38}$$

, where $G \in \mathbb{R}^{N\times N}$ is symmetric,$c \in \mathbb{R}^N$ and the vectors of lower and upper bounds $l, u \in (\mathbb{R} \cup \{-\infty, \infty\})^N$, with $l < u$. The gradient of the objective function F gets denoted with

$$g = \nabla F = Gx + c.$$

3.4.1.1 Generalized cauchy point

Let $x \in \mathbb{R}^N$ be an arbitrary feasible point. By projecting the steepest decent direction onto the box defined by l and u we get the piecewise linear path

$$x(t) = P(x - tg, l, u)$$

, where the projection operator P is defined as in chapter 3.3. The generalized cauchy point x^c is then defined as the first local minimizer of the function $F(x(t))$ for $t \geq 0$. To obtain the generalized cauchy point, we examine each of the line segments of the function $x(t)$. To do so, first of all, we need the values of t at which the piecewice linear path has kinks. We call these values the breakpoints and define them as

$$\bar{t}_i = \begin{cases} (x_i - u_i)/g_i & \text{if } g_i < 0 \text{ and } u_i < \infty \\ (x_i - l_i)/g_i & \text{if } g_i > 0 \text{ and } l_i > -\infty \\ \infty & \text{else} \end{cases}$$

. These are the first values of t for which the component $x(t)_i$ reaches its bound. The components of $x(t)$ can therefore be written as

$$x(t)_i = \begin{cases} x_i - tg_i & \text{if } t \leq \bar{t}_i \\ x_i - \bar{t}_i g_i & \text{else} \end{cases}$$

. Now, we eliminate duplicate and zero values of $\bar{t}_i$ to get the sorted set $\{t_1, t_2, \ldots, t_l\}$ and examine $x(t)$ in the intervals $[0, t_1]$,$[t_1, t_2]$ until $[t_{l-1}, t_l]$. Suppose we have examined all the intervals up to $[t_{j-2}, t_{j-1}]$ and have not identified a local minimizer. For the interval $[t_{j-1}, t_j]$ we write the function $x(t)$ as

$$x(t) = x(t_{j-1}) + \delta t p^{j-1}$$

, where $\delta t = t - t_{j-1} \in [0, t_j - t_{j-1}]$ and the vector p^{j-1} is defined as

$$p_i^{j-1} = \begin{cases} -g_i & \text{if } t_{j-1} < \bar{t}_i \\ 0 & \text{else} \end{cases}$$

. Now, we can write the objective function F on the given line segment as

$$F(x(t)) = F_{j-1} + F'_{j-1}\delta t + \frac{1}{2}F''_{j-1}\delta t^2 \tag{39}$$

, where the coefficients F_{j-1},F'_{j-1} and F''_{j-1} are defined as

$$\begin{aligned} F_{j-1} &= c^\intercal x(t_{j-1}) + \frac{1}{2}x(t_{j-1})^\intercal G x(t_{j-1}) \\ F'_{j-1} &= c^\intercal p^{j-1} + x(t_{j-1})^\intercal G p^{j-1} \\ F''_{j-1} &= \left(p^{j-1}\right)^\intercal G p^{j-1} \end{aligned}$$

. Differentiating (39) with respect to δt and setting it to zero gives $\delta t^* = -F'_{j-1}/F''_{j-1}$. Now, the following three cases can occur:

$F'_{j-1} > 0$	local minimizer at	$t = t_{j-1}$
$\delta t^* \in [0, t_j - t_{j-1})$	local minimizer at	$t = t_{j-1} + \delta t^*$
else	move on to the next interval	

An algorithm for the computation of the generalized cauchy point is therefore straight forward to implement by simply iterating over all the intervals and stopping with the

correct minimizer, if one of the first two cases occurs, or stopping with $x(t_l)$, if we have reached the last interval without finding a local minimizer. While executing this algorithm, computational savings can be made due to Conn et al. (1988) by rather updating the coefficients F'_{j-1} and F''_{j-1} then calculating them from scratch. For the update of F'_{j-1} one can use

$$x(t_j) = x(t_{j-1}) + (t_j - t_{j-1})p^{j-1}$$

and then calculate

$$\begin{aligned}
F'_j &= c^\intercal p^j + x(t_j)^\intercal G p^j \\
&= c^\intercal p^j + x(t_{j-1})^\intercal G p^j + (t_j - t_{j-1}) \left(p^{j-1}\right)^\intercal G p^j \\
&= c^\intercal p^{j-1} + c^\intercal \left(p^j - p^{j-1}\right) + x(t_{j-1})^\intercal G p^{j-1} + x(t_{j-1})^\intercal G \left(p^j - p^{j-1}\right) \\
&+ (t_j - t_{j-1}) \left(p^{j-1}\right)^\intercal G p^{j-1} + (t_j - t_{j-1}) \left(p^{j-1}\right)^\intercal G \left(p^j - p^{j-1}\right) \\
&= F'_{j-1} + c^\intercal \tilde{p}^j + x(t_{j-1})^\intercal G \tilde{p}^j + (t_j - t_{j-1}) F''_{j-1} + (t_j - t_{j-1}) \left(p^{j-1}\right)^\intercal G \tilde{p}^j \\
&= F'_{j-1} + c^\intercal \tilde{p}^j + x(t_j)^\intercal G \tilde{p}^j + (t_j - t_{j-1}) F''_{j-1}
\end{aligned}$$

, where $\tilde{p}^j = p^j - p^{j-1}$. For the update of F''_{j-1} one again calculates

$$\begin{aligned}
F''_j &= \left(p^j\right)^\intercal G p^j \\
&= \left(p^j\right)^\intercal G p^{j-1} + \left(p^j\right)^\intercal G \left(p^j - p^{j-1}\right) \\
&= \left(p^{j-1}\right)^\intercal G p^{j-1} + \left(p^j - p^{j-1}\right)^\intercal G p^{j-1} + \left(p^j\right)^\intercal G \left(p^j - p^{j-1}\right) \\
&= F''_{j-1} + \left(p^{j-1} + p^j\right)^\intercal G \tilde{p}^j \\
&= F''_{j-1} + \left(\tilde{p}^j + 2p^{j-1}\right)^\intercal G \tilde{p}^j
\end{aligned}$$

. The matrix vector product $b = G\tilde{p}^j$ can be evaluated efficiently. Since the vectors p^{j-1} and p^j differ in just a few components, usually only one, it can be written as

$$b = \sum_{\tilde{p}_i^j \neq 0} G_i \tilde{p}_i^j$$

, where G_i denotes the i-th column of G, and therefore the updates

$$\begin{aligned}
F'_j &= F'_{j-1} + (t_j - t_{j-1}) F''_{j-1} + c^\intercal \tilde{p}^j + x(t_j)^\intercal b \\
F''_j &= F''_{j-1} + \left(\tilde{p}^j + p^{j-1}\right)^\intercal b
\end{aligned}$$

can be done in only additional three vector-vector products.

3.4.1.2 Subspace minimization

After the computation of the generalized cauchy point, we define the working set $\mathcal{W}$ as

$$\mathcal{W} = \{i : x_i^c = l_i \text{ or } x_i^c = u_i\}$$

and approximately solve the reduced problem

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} x^\intercal G x + x^\intercal c \quad \text{subject to} \begin{cases} x_i = x_i^c & \forall i \in \mathcal{W} \\ l_i \leq x_i \leq u_i & \forall i \notin \mathcal{W} \end{cases} \tag{40}$$

. To do so, note that this problem has the form

$$\min_{x \in \mathbb{R}^N} \frac{1}{2} x^\intercal G x + x^\intercal c \quad \text{subject to } Ax = x_r^c \tag{41}$$

, if we ignore the bound-constraints and define the column vector $x_r^c \in \mathbb{R}^{|\mathcal{W}|}$ as the reduced cauchy point, which consists only of the entries of x^c, which are in the working set, i.e.

$$x_r^c = (x_i^c)_{i \in \mathcal{W}}$$

. The matrix $A \in \mathbb{R}^{|\mathcal{W}| \times N}$ has a very simple form since the rows of A are given by the canonical basis vectors $e_i^\intercal$, with $i \in \mathcal{W}$. A matrix Z that contains a basis of the null space of A in its columns is given by $Z = (e_i)_{i \notin \mathcal{W}}$. If we define $Y = A^\intercal$ we have that the matrix $[Y\ Z]$ is nonsingular since it is the identity matrix with permuted columns. Therefore, we can write any solution x of the linear system $Ax = x_r^c$ in the form

$$x = Y x_y + Z x_z$$

for some $x_y \in \mathbb{R}^{|\mathcal{W}|}$ and $x_z \in \mathbb{R}^{N - |\mathcal{W}|}$. Now, we have that

$$x_r^c = Ax = AY x_y = x_y$$

. Any feasible x for problem (41) can therefore be written as

$$x = Y x_r^c + Z x_z$$

, for some $x_z \in \mathbb{R}^{N-|\mathcal{W}|}$. Problem (41) therefore reduces to

$$\min_{x_z \in \mathbb{R}^{N-|\mathcal{W}|}} \frac{1}{2} x_z^\intercal Z^\intercal G Z x_z + x_z^\intercal c_z \tag{42}$$

, with $c_z = Z^\intercal G Y x_y + Z^\intercal c$. Suppose for the moment that $Z^\intercal G Z$ is positive definite. Problem (42) can then be solved with the preconditioned conjugate gradient method (see Nocedal et Wright 2006 [p.460]). Since the aim is to approximately solve problem (40) though and we can in general not assume that $Z^\intercal G Z$ is positive definite, we will incorporate the ideas of Steilhaugs conjugate gradient method (see Nocedal et Wright 2006 [p.171]) for solving the trust region subproblem into the preconditioned conjugate gradient method. That is whenever the preconditioned conjugate gradient method produces a direction of non-positive curvature along $Z^\intercal G Z$ or produces a step that lies outside the feasible box defined by l and u, the algorithm should terminate with an approximate solution that is given by $Y x_y + Z(\tilde{x}_z + \alpha_{min} \tilde{d}_z)$, where $\tilde{x}_z$ is the current iterate of the preconditioned conjugate gradient method, $\tilde{d}_z$ is the current search direction and α_{min} is the largest stepsize such that $l \leq Y x_y + Z(\tilde{x}_z + \alpha_{min} \tilde{d}_z) \leq u$. Since the components of the vector $Y x_y + Z(\tilde{x}_z + \alpha_{min} \tilde{d}_z)$ are given by

$$\left(Y x_y + Z(\tilde{x}_z + \alpha_{min} \tilde{d}_z) \right)_i = \begin{cases} (x_y)_i & \text{if } i \in \mathcal{W} \\ (\tilde{x}_z + \alpha_{min} \tilde{d}_z)_i & \text{if } i \notin \mathcal{W} \end{cases}$$

, we only have to choose α_{min} as the largest stepsize such that $l_r \leq \tilde{x}_z + \alpha_{min} \tilde{d}_z \leq u_r$, where again l_r and u_r are the reduced lower and upper bounds, but now with respect to $\mathcal{W}^C$, defined as

$$l_r = (l_i)_{i \notin \mathcal{W}} \quad u_r = (u_i)_{i \notin \mathcal{W}}$$

. The computation of α_{min} can be done by

$$\alpha_{min} = \min_{i \in \{1, \ldots, N-|\mathcal{W}|\}} \alpha_i$$

, where

$$\alpha_i = \begin{cases} \left((l_r)_i - (\tilde{x}_z)_i \right) / (\tilde{d}_z)_i & \text{if } (\tilde{d}_z)_i < 0 \text{ and } (l_r)_i > -\infty \\ \left((u_r)_i - (\tilde{x}_z)_i \right) / (\tilde{d}_z)_i & \text{if } (\tilde{d}_z)_i > 0 \text{ and } (u_r)_i < \infty \\ \infty & \text{else} \end{cases}$$

. The algorithm for approximately solving the reduced problem (40) can be specified as

Data: starting point x_z, convergence tolerance η, preconditioner W_{zz}
Result: approximate solution x^*
Set $r_z = Z^\intercal G Z x_z + c_z$, $g_z = W_{zz}^{-1} r_z$ and $d_z = -g_z$;
while $r_z^\intercal g_z > \eta$ **do**
  if $d_z^\intercal Z^\intercal G Z d_z \leq 0$ **then**
    Compute α_{min} with $\tilde{x}_z = x_z$ and $\tilde{d}_z = d_z$;
    $x_z = x_z + \alpha_{min} d_z$;
    break
  end
  $\alpha = r_z^\intercal g_z / d_z^\intercal Z^\intercal G Z d_z$;
  $x_z^{old} = x_z$;
  $x_z = x_z + \alpha d_z$;
  if $l_r \leq x_z \leq u_r$ *not satisfied* **then**
    Compute α_{min} with $\tilde{x}_z = x_z^{old}$ and $\tilde{d}_z = d_z$;
    $x_z = x_z^{old} + \alpha_{min} d_z$;
    break
  end
  $r_z^+ = r_z + \alpha Z^\intercal G Z d_z$;
  $g_z^+ = W_{zz}^{-1} r_z^+$;
  $\beta = (r_z^+)^\intercal g_z^+ / r_z^\intercal g_z$;
  $d_z = -g_z^+ + \beta d_z$;
  $g_z = g_z^+$;
  $r_z = r_z^+$;
end
return $x^* = Y x_r^c + Z x_z$;

Algorithm 4: Preconditioned CG for Reduced Problem

Since the aim is, to produce a step that has a function value lower equal to the function value of the generalized cauchy point, we choose the starting column vector $x_z \in \mathbb{R}^{N-|\mathcal{W}|}$ to be

$$x_z = (x^c)_{i \notin \mathcal{W}}$$

, since then we have

$$x^c = Yx_y + Zx_z$$

, and we start the subspace minimization at the generalized cauchy point. Note that due to the simple form of Y and Z the matrix $Z^\intercal GZ$ can easily be computed by removing the rows and columns indexed by $\mathcal{W}$ and the vector x^* can be obtained by

$$x_i^* = \begin{cases} x_i^c & i \in \mathcal{W} \\ (x_z)_i & i \notin \mathcal{W} \end{cases}$$

. Since algorithm 4 only needs one matrix vector product for each iteration, if implemented correctly, it is suitable for large scale problems. A general framework of the gradient projection method for quadratic programs can now be specified as (see Nocedal et Wright 2006 [p.489]):

Data: feasible starting point x^0
Result: approximate solution x^*
for $k = 0, 1, 2, \ldots$ **do**
  if x^k *satisfies the KKT conditions of problem (38)* **then**
    return $x^* = x^k$;
  end
  Set $x = x^k$ and compute the cauchy point x^c;
  Compute approximate solution x^+ of the reduced problem with algorithm 4;
  $x^{k+1} = x^+$;
end

Algorithm 5: Gradient Projection Method for QPs

3.4.2 Trust region based gradient projection method for general bound-constrained nonlinear programs

To solve general bound-constrained problems like (25), a trust region based algorithm is used instead of the procedure in algorithm 5. The algorithm that gets presented follows the descriptions in Nocedal et Wright 2006 [p.554-556]. The l-th iterate of this algorithm is denoted by x^{k_l}, where $k_0 = k$, to avoid confusion with the iterates x^k in algorithm

3. Suppose we are now in the l-th iteration of this algorithm. The current trust region radius is denoted by Δ_l. The second order approximation of $\mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k)$ is given by

$$m^l(x^{k_l} + s^l) = \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) + (s^l)^\intercal \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) + \frac{1}{2}(s^l)^\intercal \nabla^2_{xx} \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) s^l$$

. If we now define

$$\begin{aligned} G^l &= \nabla^2_{xx} \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) \\ c^l &= \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) \end{aligned}$$

, the l-th trust region subproblem is given by

$$\min_{x \in \mathbb{R}^N} \frac{1}{2}(s^l)^\intercal G s^l + (s^l)^\intercal c \quad \text{subject to } \begin{cases} l \leq x^{k_l} + s^l \leq u \\ ||s^l||_\infty \leq \Delta_l \end{cases} \tag{43}$$

. Note that the infinity norm is used to define the trust region, since the constraints of s^l in problem (43) are then given by

$$\begin{array}{rcccl} l - x^{k_l} & \leq & s^l & \leq & u - x^{k_l} \\ -\Delta_l \mathbb{1}_N & \leq & s^l & \leq & \Delta_l \mathbb{1}_N \end{array}$$

and can be combined to

$$\max(l - x^{k_l}, -\Delta_l \mathbb{1}_N) \leq s^l \leq \min(u - x^{k_l}, \Delta_l \mathbb{1}_N)$$

, where max and min have to be understood componentwise. Problem (43) therefore reduces to

$$\min_{x \in \mathbb{R}^N} \frac{1}{2}(s^l)^\intercal G s^l + (s^l)^\intercal c \quad \text{subject to } \hat{l}_l \leq s^l \leq \hat{u}_l \tag{44}$$

, where $\hat{l}_l = \max(l - x^{k_l}, -\Delta_l \mathbb{1}_N)$ and $\hat{u}_l = \min(u - x^{k_l}, \Delta_l \mathbb{1}_N)$. A step s^l can now be found by first computing the generalized cauchy point x^c, staring at $s^l = 0$, and then using algorithm 4 to compute an approximate solution of the reduced problem. After that the update of x^{k_l} and the trust region radius Δ_l can be done in the usual trust region step quality measurement procedure. A general framework of the algorithm can be specified as:

Data: feasible starting point $x^{k_0} = x^k$, convergence tolerance ω_k
Result: approximate solution x^{k+1}
Set $0 < \underline{\omega} < 1 < \overline{\omega}$ and $0 < \underline{\mu} < 1 < \overline{\mu}$;
Set $l = 0$;
Evaluate $G^0 = \nabla^2_{xx}\mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)$;
Evaluate $c^0 = \nabla_x \mathcal{L}_A(x^k, \bar{\lambda}^k; \theta_k)$;
while $||x^{k_l} - P(x^{k_l} - \nabla_x \mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k), l, u)|| > \omega_k$ **do**
    Compute $\hat{l}_l$;
    Compute $\hat{u}_l$;
    Compute generalized cauchypoint x^c starting at $s^l = 0$ of problem (44);
    Compute step s^l with algorithm 4 using x^c;
    Compute ared= $\mathcal{L}_A(x^{k_l}, \bar{\lambda}^k; \theta_k) - \mathcal{L}_A(x^{k_l} + s^l, \bar{\lambda}^k; \theta_k)$;
    Compute pred= $-(c^l)^\intercal s^l - 1/2(s^l)^\intercal G s^l$;
    Compute reduction measure μ = ared/pred;
    if $\mu < \underline{\mu}$ **then**
        $x^{k_{l+1}} = x^{k_l}$;
        $\Delta_{l+1} = \underline{\omega}\Delta_l$;
    end
    if $\underline{\mu} \leq \mu \leq \overline{\mu}$ **then**
        $x^{k_{l+1}} = x^{k_l} + s^l$;
        $\Delta_{l+1} = \Delta_l$;
        Evaluate $G^{l+1} = \nabla^2_{xx}\mathcal{L}_A(x^{k_{l+1}}, \bar{\lambda}^k; \theta_k)$;
        Evaluate $c^{l+1} = \nabla_x \mathcal{L}_A(x^{k_{l+1}}, \bar{\lambda}^k; \theta_k)$;
    end
    if $\mu > \overline{\mu}$ **then**
        $x^{k_{l+1}} = x^{k_l} + s^l$;
        $\Delta_{l+1} = \overline{\omega}\Delta_l$;
        Evaluate $G^{l+1} = \nabla^2_{xx}\mathcal{L}_A(x^{k_{l+1}}, \bar{\lambda}^k; \theta_k)$;
        Evaluate $c^{l+1} = \nabla_x \mathcal{L}_A(x^{k_{l+1}}, \bar{\lambda}^k; \theta_k)$;
    end
    $l = l + 1$;
end

Algorithm 6: Nonlinear Gradient Projection Method

The variables x^k for the starting point, x^{k+1} for the approximate solution and ω_k for

the convergence tolerance are used to denote the relationship to algorithm 3. Also, for this reason the function $\mathcal{L}_A$ is used in the algorithm instead of a general objective function F. The convergence analysis of algorithm 6 is done in the context of a general bound-constrained problem

$$\min_{x \in \mathbb{R}^N} F(x) \quad \text{subject to } l \leq x \leq u \tag{45}$$

, though and the iterates x^{k_l} are replaced by x^l. The following theorem without proof is a summary of the theory presented in Conn et al. (1986).

Theorem 12. *Let x^0 be a feasible starting point for algorithm 6 of problem (45). Suppose that the set*

$$L = \{x \in \mathbb{R}^N : F(x) \leq F(x^0)\} \cap \{l \leq x \leq u\}$$

is compact and has non empty interior. Further suppose that the objective function F is twice continuously differentiable. Then the iterates x^l of algorithm 6 fulfill

$$\lim_{l \to \infty} x^l - P(x^l - \nabla F(x^l), l, u) = 0$$

. Suppose further, that for every accumulation point $x^* \in L$ of the sequence $(x^l)_{l \in \mathbb{N}}$ the strict complementary condition

$$i \in \mathcal{A}(x^*) \Rightarrow |\nabla F(x^*)_i| > 0 \tag{46}$$

holds. Then, for every subsequence $(x_{l_j})_{j \in \mathbb{N}}$ that converges to $x^* \in L$ it holds that x^* is a KKT point of problem (45) and that

$$\mathcal{A}(x_{l_j}) = \mathcal{A}(x^*)$$

for sufficiently larger j. This implies, that if the complete sequence x^l converges to a single limit point x^*, the correct active set is identified after finitely many iterations.

According to Conn et al., the assumption on the set L only says that problem (45) is non trivial and cannot be reduced to a lower-dimensional problem. Concerning the model, we can assume that this condition is always satisfied. Note that assumption (46)

is crucial, since we cannot guarantee that it is always satisfied and if it does not hold the working set $\mathcal{W}$ may not settle down to the correct active set. Although we have that

$$\lim_{l\to\infty} x^l - P(x^l - \nabla F(x^l), l, u) = 0$$

, algorithm 6 may fail in the degenerate case. Note also, that the general theory presented in Conn et al. (1986) does not give an algorithm for finding the step s^l and rather requires of a general step s^l that the following holds

$$\begin{aligned} F(x^l) - m^l(x^l + s^l) &\geq \beta_1 \left(F(x^l) - m^l(x^l + x^c) \right) \\ ||s^l|| &\leq \Delta_l \\ \mathcal{A}(x^c) &\subseteq \mathcal{A}(x^l + s^l) \end{aligned}$$

, for $\beta_1 \in (0, 1]$ and where x^c is the generalized cauchy point in iteration l. The step produced by algorithm 4 satisfies these conditions with $\beta_1 = 1$, if we start at the correct x_z described in chapter 3.4.1.2. In the implementation I have encountered situations in which due to numerical reasons the first condition is violated. Therefore, the predicted reduction of the generalized cauchy point is compared to the predicted reduction of the step produced by algorithm 4 in every iteration and if the first condition is violated we set $s^l = x^c$. Other situations that can cause difficulties for the algorithm are described in Conn et al. (2010). According to Conn et al., variables or constraints, whose numerical values are widely different in magnitude, should be avoided. Therefore the definition of the capacity-units motivated in chapter 2.2 is a little more delicate. The measure should be chosen in a way that $L \approx 1$, since this leads to slack-variables that are approximately of the same size as the remaining variables.

4 Implementation and results

The implementation of the algorithms described in the last chapter is done in MATLAB®. An overview of the implemented functions can be seen in figure 2. The MainAlgorithm.m file contains the bound constrained Lagrangian method. With a preconditioning flag one can turn the preconditioning with an inexact modified Cholesky preconditioner on or off, which uses a modified incomplete Cholesky decomposition. The algorithm can be found in Nocedal et Wright 2006 [p.174]. It is possible to print information of convergence progress in a pop up window with an information flag, like, for example, the current constraint violation. Once the MainAlgorithm.m terminates with a solution, the func-

tion GetKKT.m calculates the Lagrange multipliers for the inequality constraints and prints the first order optimality measure, which is the norm of $\nabla_x \mathcal{L}^p$ evaluated at the calculated x and calculated Lagrange multiplier vectors. After that the function checkSOSC.m checks if the second order sufficient conditions are satisfied with the algorithm described in chapter 3.1.3. It uses the function spspces.m (MathWorks®, 2006), which is specially designed for the computation of the null space of large sparse matrices.

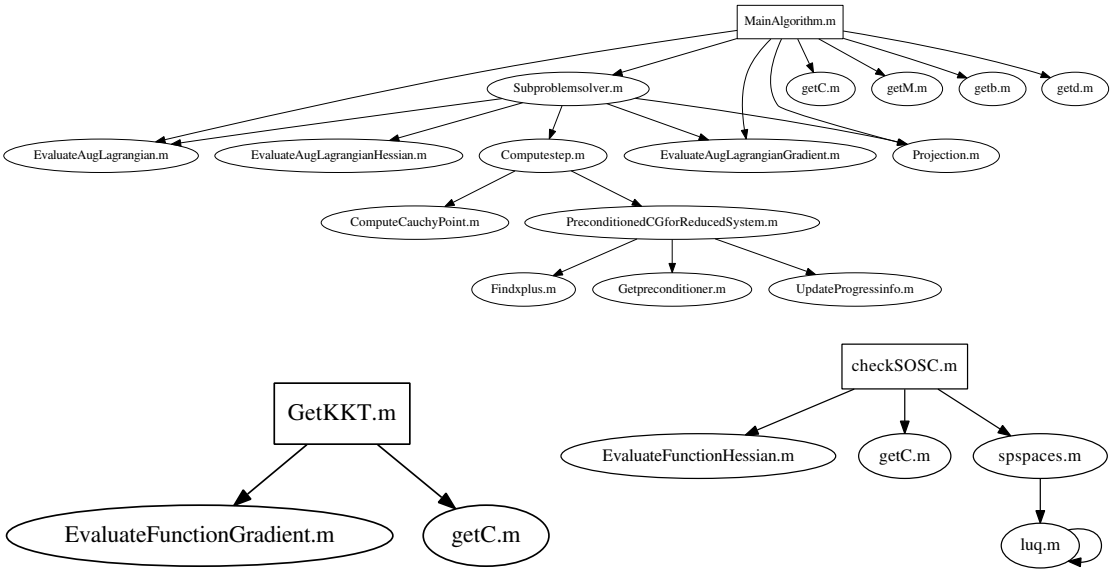


Figure 2: Overview of Implemented Functions

4.1 Impact of the parameters α,β and γ

First results of the model are presented in this chapter, especially the impact of the parameters α,β and γ will be shown in an example. For this purpose we consider the warehouse with 42 storage locations, which can be seen in figure 3.

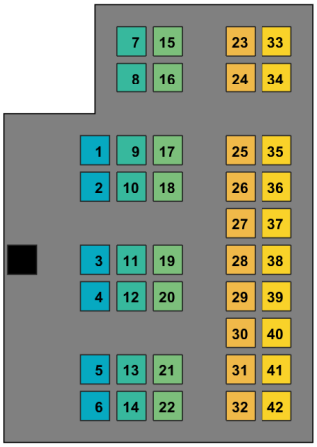


Figure 3: Warehouse

The depot is represented by the black rectangle. The values of the variables d_k are given in table 2. They are found by the Dijkstra algorithm for the calculation of the length of the shortest path from a source vertex to all the other vertices in a weighted graph (see e.g Ahuja et al. 1993) and assuming that the distance between the boxes is given by 1.

k	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
d_k	4.4721	2.2361	2.2361	2.8284	2.8284	5.3852	6.4721	5.4721	5.4721	3.2361
k	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
d_k	3.2361	3.8284	3.8284	6.3852	9.4721	8.4721	6.4721	4.2361	4.2361	4.8284
k	**21**	**22**	**23**	**24**	**25**	**26**	**27**	**28**	**29**	**30**
d_k	4.8284	7.3852	9.4721	8.4721	7.2361	6.2361	5.2361	6.2361	6.8284	5.8284
k	**31**	**32**	**33**	**34**	**35**	**36**	**37**	**38**	**39**	**40**
d_k	6.8284	7.8284	12.4721	11.4721	11.4721	12.4721	13.4721	14.4721	15.3852	14.3852
k	**41**	**42**								
d_k	13.3852	12.3852								

Table 2: Distances to the Depot

Suppose we have 4 articles to be stored. The storage location capacity $L = 1$ and the

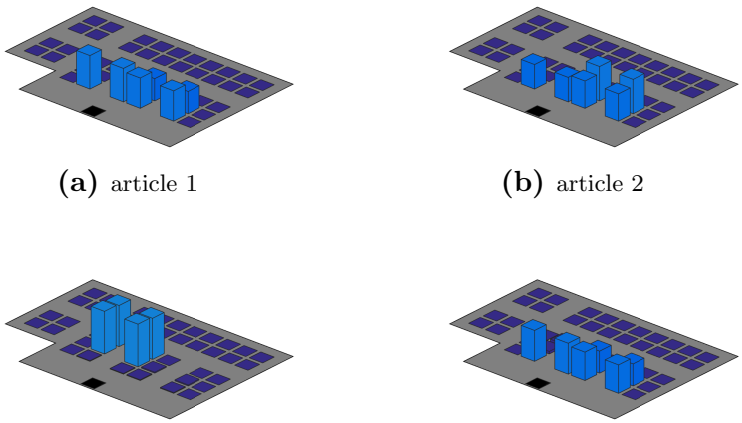
parameter $\epsilon = 0.0001$. The article correlations λ_{ij} are given by the matrix

$$\begin{pmatrix} \times & 0.5 & 0.01 & 0.9 \\ \times & \times & 0.1 & 0.4 \\ \times & \times & \times & 0.2 \\ \times & \times & \times & \times \end{pmatrix}$$

and the values of μ_i and g_i are given by

$$\begin{aligned} \mu_1 &= 0.4 \quad g_1 = 2 \\ \mu_2 &= 0.3 \quad g_2 = 3 \\ \mu_3 &= 0.2 \quad g_3 = 4 \\ \mu_4 &= 0.1 \quad g_4 = 1 \end{aligned}$$

. To begin with, we calculate the optimal probability distributions for the articles with the parameters $\alpha = \beta = \gamma = 1$. The solution can be seen in figure 4.



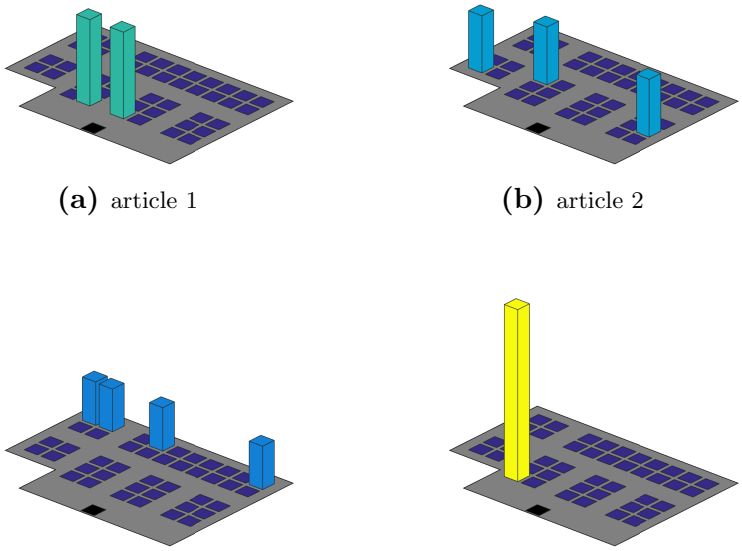
(a) article 1 **(b)** article 2

(c) article 3 **(d)** article 4

Figure 4: Test Result for $\alpha = \beta = \gamma = 1$

As expected, article 1 and article 4 have nearly the same distribution. This is due to the fact that with $\lambda_{14} = 0.9$ we have the highest correlation between them. The two next largest values $\lambda_{12} = 0.5$ and $\lambda_{24} = 0.4$ force the distribution of article 2 to be near to the one of article 1 and 4. Since the lowest correlations are the ones concerning article 3, there is no need to store article 3 next to the other articles. With regard to the proximity to the depot, we can see that the distributions of article 1,2 and 3 are sorted

according to their decreasing values of μ_i. Although $\mu_4 = 0.1$ is the lowest value, we see that the distribution of article 4 is concentrated near the depot. This is due to the fact that article 4 needs with $g_4 = 1$ the least space to be stored and can therefore fill up the remaining storage space. Next we have a look at the three extreme cases, where α,β and γ are respectively much higher than the remaining two parameters. In figure 5 we can see the result for $\alpha = \beta = 1$ and $\gamma = 100$.

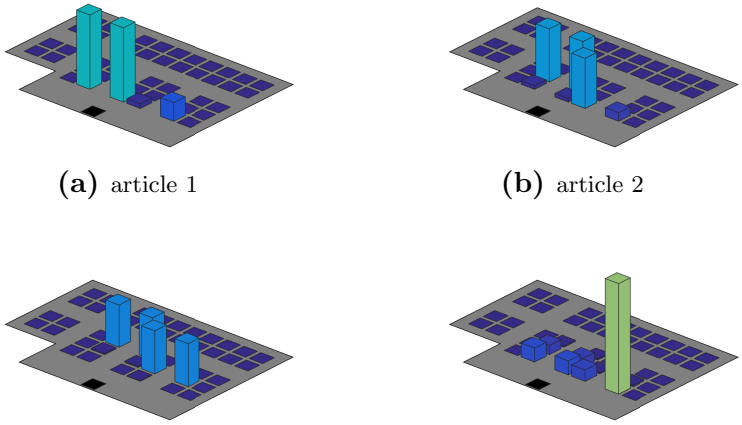


(a) article 1 **(b)** article 2

(c) article 3 **(d)** article 4

Figure 5: Test Result for $\alpha = \beta = 1$ and $\gamma = 100$

As we can see, the value of γ forces the distributions to be clustered as much as possible. According to their values of g_i and due the storage location capacity constraint, the distribution of article 4 is concentrated in a single storage location, the distribution of article 1 is concentrated in two storage locations, and the distributions of article 2 and 3 are respectively concentrated in three and four storage locations. The factors article correlation and article turnover are nearly neglected. Now, we have a look at the case in which $\alpha = \gamma = 1$ and $\beta = 100$.

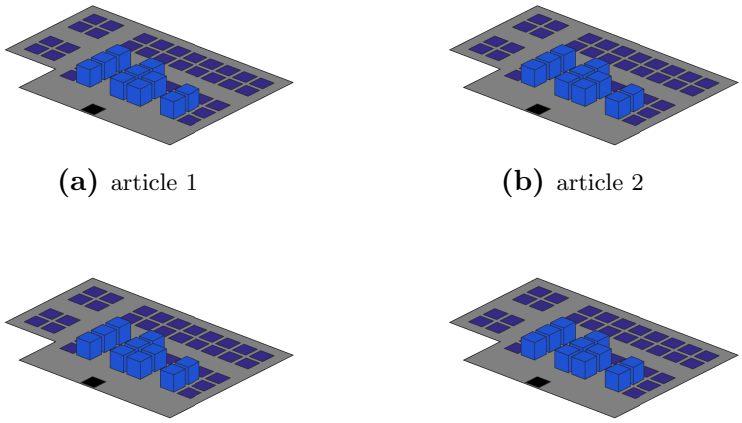


(a) article 1 **(b)** article 2

(c) article 3 **(d)** article 4

Figure 6: Test Result for $\alpha = \gamma = 1$ and $\beta = 100$

The distributions of article 1,2 and 3 are again sorted according to their decreasing values of μ_i, but now with a higher degree of severity. Again, we can see the effect of g_4 and how article 4 is able to fill up the remaining space near the depot. The last extreme case can be seen in figure 7, with $\alpha = 50$ and $\beta = \gamma = 1$. As expected, all the articles are concentrated in the same storage locations and are uniformly distributed among those, since we put all the emphasis on storing articles that are somehow correlated close to each other. From this we can also see that α and γ are somehow direct opponents. While an increasing parameter α pushes the distributions to the uniform distribution, the parameter γ induces the formation of small clusters in the warehouse.



(a) article 1 **(b)** article 2

(c) article 3 **(d)** article 4

Figure 7: Test Result for $\beta = \gamma = 1$ and $\alpha = 50$

4.2 Testing of the model

To give a first impression of how the storage location assignment, according to the calculated distributions, affects the costs in form of travel distance for incoming and outgoing articles in comparison to the random storage policy, I implemented a simulation of a working year with 200 working days in the above warehouse. For the simulation we suppose that we have 100 articles to be stored and commissioned. Each article has a daily demand following a normal distribution with a given mean and standard deviation. From the daily demands of each of the articles a total number of 57654 commissions are created, with a minimum number of 269 and a maximal number 313 commissions per day. The commissions are created in a way that the first and the last twenty of the articles are more likely commissioned together than the rest. The maximum number of articles per commission is 10 and the commission size is uniformly distributed between 1 and 10. The inventory is controlled according to the basic economic order quantity model (see Heizer et Render 2011), i.e. we have a fixed economic order quantity (EOQ) and a fixed reorder point (ROP) for each article given by the formulas

$$\mathrm{EOQ}_i = \sqrt{\frac{2D_iS_i}{H_i}}$$
$$\mathrm{ROP}_i = d_i \cdot l_i$$

where D_i is the annual demand of article i, S_i are the reordering fix costs for article i, H_i are the holding costs per SKU per year for article i, d_i is the average daily demand of article i and l_i is the lead time of article i. For convenience we assume, that the all the articles have the same article capacity of 0.005, so the values g_i are given by the average inventory level times 0.005, i.e.

$$g_i = \frac{\text{EOQ}_i}{2} \cdot 0.005$$

. We also assume that $S_i = H_i = l_i = 1$ for all $i \in \{1, \ldots, 100\}$. A detailed list of all the involved values can be found in table 3 in the appendix. The values of λ_{ij} are computed according to the created commissions. For the algorithm the parameters $\epsilon = 0.0001$,$\alpha = 3$,$\beta = 2$ and $\gamma = 0.005$ are used. An overview of the simulation is given in figure 8.

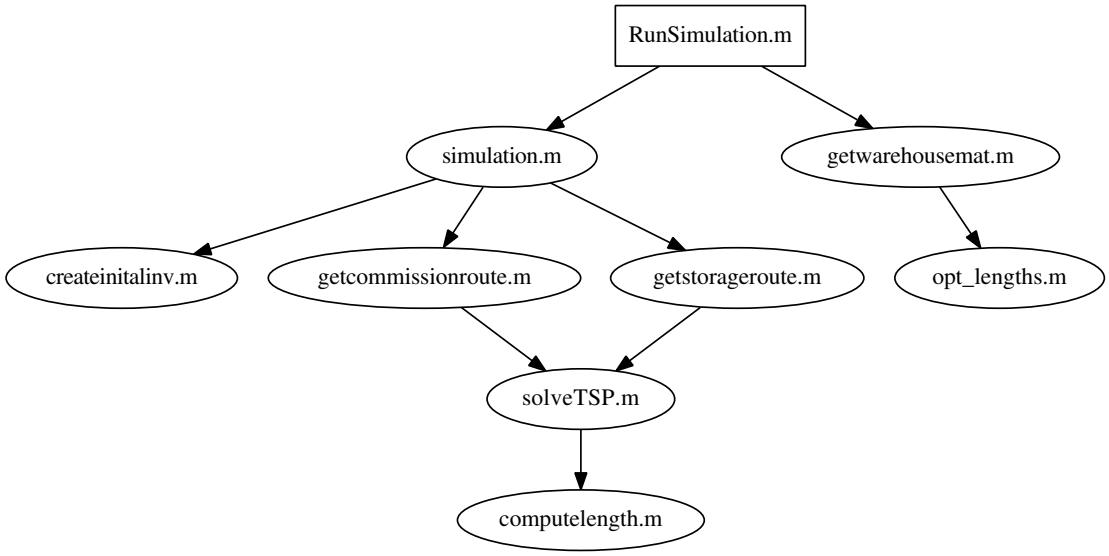


Figure 8: Overview of Implemented Functions for the Simulation

The RunSimulation.m file gets the desired probability distributions as an input, i.e. the calculated distributions or uniform distributions. It starts to build up an initial inventory with the function createinitialinv.m. This is done by storing an amount of $\text{EOQ}_i/2$ SKUs of each article according to the given probability distribution. After that the simulation starts and the program loops through the working days. At the beginning of each working day the function getstorageroute.m stores the incoming SKUs according to the given probability distribution and computes the length of the optimal storage route by solving the resulting travelling salesman problem with the function solveTSP.m. After

that, the program loops through all the commissions of the day and computes the lengths of the optimal commission routes in the function getcommissionroute.m. Since each of the articles in a commission is possibly stored in several storage locations, a policy for the choice of the storage location from which a given article should be picked has to be introduced. The implementation uses the First In First Out (FIFO) policy, since this is common practice (see Van den Berg 1999). At the end of each working day, the program checks if the inventory level of an article is lower or equal to the corresponding reorder point and triggers an order if necessary. The amount of SKUs that get ordered is given by EOQ_i and the replenishment order arrives at the warehouse after l_i working days. The function solveTSP.m uses the state of the art TSP solver concorde which is freely available for academic use (see Applegate et al. 2006). The outputs of the simulation are the total travel distances for incoming and outgoing goods. The results are summarized in figure 9.

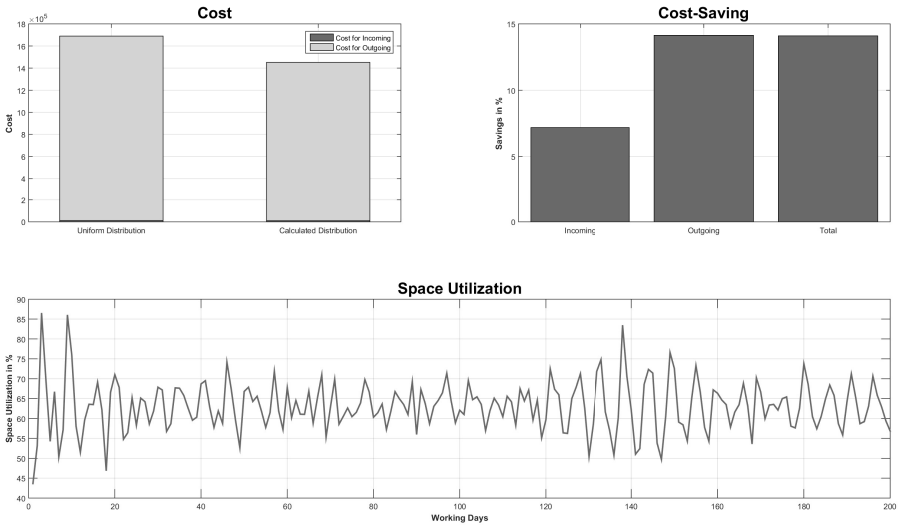


Figure 9: Simulation Results

One can see, that the storage location assignment according to the calculated distributions saves nearly 15% of total travel distance, while the space utilization remains unchanged.

5 Conclusion and future work

The simulation results indicate the advantages of the model. The benefit of high space utilization of the random storage policy gets extended by three important factors that can reduce travel distance in the storing and retrieving process. I have noticed, that the convergence speed heavily depends on the choice of the parameters α,β and γ and that sometimes the algorithm struggles with the identification of the correct active set, which is, according to my opinion with regard to the presented theory due to degeneracy. The choice of the parameters α,β and γ is up to now by trial and error and I also want to mention that, in general, we can not hope for a unique minimizer, although this might be desirable. Therefore, a first step towards further development of the model has to be to automate the choice of the parameters using the given input data. I also think that it might be possible to calculate a proportion between α and γ such that the objective function becomes at least pseudoconvex. If this is doable, the quasiconvexity of the inequality constraints and the linearity of the equality constraints would ensure uniqueness of the minimizer. A second step would be to trace the parameters λ_{ij} and μ_i with time series analysis tools, since this would lead to a dynamical adjustment to changing circumstances, like, for example, periodic demand. One could freeze the parameters λ_{ij} and μ_i for a given period of time and calculate the optimal probability distributions with the presented model. After that, the storage location assignment is made according to the calculated distributions, until the given period ends. Afterwards, an update of λ_{ij} and μ_i takes place and the distributions get recomputed. To do so, the length of this period has to be analyzed previously. The timeframe has to be long enough to ensure that the turnover is sufficient to rearrange the articles in the warehouse and short enough to prevent the parameters λ_{ij} and μ_i from changing to quickly. The case of unpredictable parameters for some articles could be handled by fixing the probability distribution of those articles to the uniform distribution, until the parameters are stable enough to reenable the automated calculation of their distributions. A third step should of course be to do further tests in different scenarios with different storing and retrieving techniques and to incorporate disregarded factors in the model.

6 Appendix

article	demand ∅	demand σ	article-cap.	D_i	d_i	S_i	H_i	l_i	EOQ_i	ROP_i	μ_i	g_i
1	35	0.5	0.005	7008	35.04	1	1	1	119	36	0.023116	0.2975
2	32	0.5	0.005	6406	32.03	1	1	1	114	33	0.02113	0.285
3	34	0.5	0.005	6804	34.02	1	1	1	117	35	0.022443	0.2925
4	34	0.5	0.005	6785	33.925	1	1	1	117	34	0.02238	0.2925
5	34	0.5	0.005	6808	34.04	1	1	1	117	35	0.022456	0.2925
6	33	0.5	0.005	6591	32.955	1	1	1	115	33	0.02174	0.2875
7	35	0.5	0.005	7005	35.025	1	1	1	119	36	0.023106	0.2975
8	32	0.5	0.005	6396	31.98	1	1	1	114	32	0.021097	0.285
9	34	0.5	0.005	6795	33.975	1	1	1	117	34	0.022413	0.2925
10	35	0.5	0.005	6986	34.93	1	1	1	119	35	0.023043	0.2975
11	33	0.5	0.005	6602	33.01	1	1	1	115	34	0.021776	0.2875
12	33	0.5	0.005	6597	32.985	1	1	1	115	33	0.02176	0.2875
13	31	0.5	0.005	6184	30.92	1	1	1	112	31	0.020398	0.28
14	31	0.5	0.005	6204	31.02	1	1	1	112	32	0.020464	0.28
15	31	0.5	0.005	6213	31.065	1	1	1	112	32	0.020493	0.28
16	32	0.5	0.005	6412	32.06	1	1	1	114	33	0.02115	0.285
17	35	0.5	0.005	7005	35.025	1	1	1	119	36	0.023106	0.2975
18	35	0.5	0.005	6998	34.99	1	1	1	119	35	0.023083	0.2975
19	31	0.5	0.005	6189	30.945	1	1	1	112	31	0.020414	0.28
20	35	0.5	0.005	6994	34.97	1	1	1	119	35	0.023069	0.2975
21	13	0.5	0.005	2601	13.005	1	1	1	73	14	0.0085793	0.1825
22	11	0.5	0.005	2213	11.065	1	1	1	67	12	0.0072995	0.1675
23	13	0.5	0.005	2597	12.985	1	1	1	73	13	0.0085661	0.1825

24	11	0.5	0.005	2192	10.96	1	1	1	67	11	0.0072302	0.1675
25	14	0.5	0.005	2808	14.04	1	1	1	75	15	0.0092621	0.1875
26	15	0.5	0.005	2991	14.955	1	1	1	78	15	0.0098657	0.195
27	15	0.5	0.005	2998	14.99	1	1	1	78	15	0.0098888	0.195
28	15	0.5	0.005	3002	15.01	1	1	1	78	16	0.009902	0.195
29	15	0.5	0.005	3008	15.04	1	1	1	78	16	0.0099218	0.195
30	15	0.5	0.005	2997	14.985	1	1	1	78	15	0.0098855	0.195
31	14	0.5	0.005	2789	13.945	1	1	1	75	14	0.0091994	0.1875
32	13	0.5	0.005	2595	12.975	1	1	1	73	13	0.0085595	0.1825
33	12	0.5	0.005	2386	11.93	1	1	1	70	12	0.0078701	0.175
34	13	0.5	0.005	2593	12.965	1	1	1	73	13	0.0085529	0.1825
35	11	0.5	0.005	2194	10.97	1	1	1	67	11	0.0072368	0.1675
36	14	0.5	0.005	2800	14	1	1	1	75	14	0.0092357	0.1875
37	15	0.5	0.005	2997	14.985	1	1	1	78	15	0.0098855	0.195
38	12	0.5	0.005	2410	12.05	1	1	1	70	13	0.0079493	0.175
39	14	0.5	0.005	2814	14.07	1	1	1	76	15	0.0092819	0.19
40	13	0.5	0.005	2600	13	1	1	1	73	13	0.008576	0.1825
41	12	0.5	0.005	2406	12.03	1	1	1	70	13	0.0079361	0.175
42	14	0.5	0.005	2795	13.975	1	1	1	75	14	0.0092192	0.1875
43	12	0.5	0.005	2399	11.995	1	1	1	70	12	0.007913	0.175
44	14	0.5	0.005	2813	14.065	1	1	1	76	15	0.0092786	0.19
45	11	0.5	0.005	2206	11.03	1	1	1	67	12	0.0072764	0.1675
46	11	0.5	0.005	2195	10.975	1	1	1	67	11	0.0072401	0.1675
47	15	0.5	0.005	3005	15.025	1	1	1	78	16	0.0099119	0.195
48	14	0.5	0.005	2795	13.975	1	1	1	75	14	0.0092192	0.1875

49	14	0.5	0.005	2802	14.01	1	1	1	75	15	0.0092423	0.1875
50	13	0.5	0.005	2591	12.955	1	1	1	72	13	0.0085463	0.18
51	12	0.5	0.005	2394	11.97	1	1	1	70	12	0.0078965	0.175
52	12	0.5	0.005	2397	11.985	1	1	1	70	12	0.0079064	0.175
53	15	0.5	0.005	3007	15.035	1	1	1	78	16	0.0099185	0.195
54	12	0.5	0.005	2401	12.005	1	1	1	70	13	0.0079196	0.175
55	11	0.5	0.005	2216	11.08	1	1	1	67	12	0.0073094	0.1675
56	11	0.5	0.005	2202	11.01	1	1	1	67	12	0.0072632	0.1675
57	14	0.5	0.005	2797	13.985	1	1	1	75	14	0.0092258	0.1875
58	13	0.5	0.005	2605	13.025	1	1	1	73	14	0.0085925	0.1825
59	14	0.5	0.005	2808	14.04	1	1	1	75	15	0.0092621	0.1875
60	14	0.5	0.005	2805	14.025	1	1	1	75	15	0.0092522	0.1875
61	14	0.5	0.005	2792	13.96	1	1	1	75	14	0.0092093	0.1875
62	14	0.5	0.005	2787	13.935	1	1	1	75	14	0.0091928	0.1875
63	12	0.5	0.005	2399	11.995	1	1	1	70	12	0.007913	0.175
64	15	0.5	0.005	3003	15.015	1	1	1	78	16	0.0099053	0.195
65	12	0.5	0.005	2403	12.015	1	1	1	70	13	0.0079262	0.175
66	11	0.5	0.005	2191	10.955	1	1	1	67	11	0.0072269	0.1675
67	12	0.5	0.005	2405	12.025	1	1	1	70	13	0.0079328	0.175
68	15	0.5	0.005	3006	15.03	1	1	1	78	16	0.0099152	0.195
69	13	0.5	0.005	2602	13.01	1	1	1	73	14	0.0085826	0.1825
70	14	0.5	0.005	2794	13.97	1	1	1	75	14	0.0092159	0.1875
71	14	0.5	0.005	2801	14.005	1	1	1	75	15	0.009239	0.1875
72	15	0.5	0.005	2985	14.925	1	1	1	78	15	0.0098459	0.195
73	12	0.5	0.005	2403	12.015	1	1	1	70	13	0.0079262	0.175

74	13	0.5	0.005	2612	13.06	1	1	1	73	14	0.0086156	0.1825
75	11	0.5	0.005	2192	10.96	1	1	1	67	11	0.0072302	0.1675
76	12	0.5	0.005	2409	12.045	1	1	1	70	13	0.007946	0.175
77	15	0.5	0.005	3013	15.065	1	1	1	78	16	0.0099383	0.195
78	15	0.5	0.005	2991	14.955	1	1	1	78	15	0.0098657	0.195
79	14	0.5	0.005	2804	14.02	1	1	1	75	15	0.0092489	0.1875
80	12	0.5	0.005	2407	12.035	1	1	1	70	13	0.0079394	0.175
81	4	0.5	0.005	797	3.985	1	1	1	40	4	0.0026289	0.1
82	2	0.5	0.005	386	1.93	1	1	1	28	2	0.0012732	0.07
83	2	0.5	0.005	403	2.015	1	1	1	29	3	0.0013293	0.0725
84	4	0.5	0.005	794	3.97	1	1	1	40	4	0.002619	0.1
85	3	0.5	0.005	593	2.965	1	1	1	35	3	0.001956	0.0875
86	3	0.5	0.005	595	2.975	1	1	1	35	3	0.0019626	0.0875
87	2	0.5	0.005	407	2.035	1	1	1	29	3	0.0013425	0.0725
88	5	0.5	0.005	984	4.92	1	1	1	45	5	0.0032457	0.1125
89	3	0.5	0.005	588	2.94	1	1	1	35	3	0.0019395	0.0875
90	1	0.5	0.005	209	1.045	1	1	1	21	2	0.00068938	0.0525
91	3	0.5	0.005	596	2.98	1	1	1	35	3	0.0019659	0.0875
92	4	0.5	0.005	792	3.96	1	1	1	40	4	0.0026124	0.1
93	5	0.5	0.005	1001	5.005	1	1	1	45	6	0.0033018	0.1125
94	2	0.5	0.005	393	1.965	1	1	1	29	2	0.0012963	0.0725
95	3	0.5	0.005	608	3.04	1	1	1	35	4	0.0020055	0.0875
96	2	0.5	0.005	400	2	1	1	1	29	2	0.0013194	0.0725
97	4	0.5	0.005	802	4.01	1	1	1	41	5	0.0026454	0.1025
98	3	0.5	0.005	604	3.02	1	1	1	35	4	0.0019923	0.0875

99	3	0.5	0.005	612	3.06	1	1	1	35	4	0.0020187	0.0875
100	2	0.5	0.005	402	2.01	1	1	1	29	3	0.001326	0.0725

Table 3: Simulation Data

References

Ahuja R., **Magnanti T., and Orlin J. 1993**. *Network Flows: Theory, Algorithms, and Applications.* Prentice-Hall, Inc.

Applegate D., Bixby R., Chvatal V., Cook W. 2006. Concorde TSP solver. `http://www.math.uwaterloo.ca/tsp/concorde.html`. Accessed: 28.06.2017 at 4 am.

Bertsekas D. 1996. *Constrained Optimization and Lagrange Multiplier Methods (Optimization and Neural Computation Series).* Athena Scientific, 1st edition.

Bindi F., **R. M.**, **Pareschi A., and Regattieri A. 2007**. Similarity coefficients and clustering techniques for the correlated assignment problem in warehousing systems. *Proceedings of 19 th International Conference on Production Research.*

Conn A., **Gould N., and Toint P. 1986**. Global convergence of a class of trust region algorithms for optimization with simple bounds. Research Report CS-86-31, Department of Computer Sciences, University of Waterloo, Waterloo, Canada.

Conn A., **Gould N., and Toint P. 1988**. Testing a class of methods for solving minimization problems with simple bounds on the variables. *Math. Comp.*, 50:399–430.

Conn A., **Gould N., and Toint P. 1991**. A Globally Convergent Augmented Lagrangian Algorithm for Optimization With General Constraints and Simple Bounds. *SIAM Journal on Numerical Analysis*, 28(2):545–572.

Conn A., **Gould N., and Toint P. 2010**. *Lancelot: A Fortran Package for Large-Scale Nonlinear Optimization (Release A).* Springer Publishing Company, Incorporated, 1st edition.

Cover T. and Thomas J. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing).* Wiley-Interscience, 2nd edition.

Eisenstein D. 2008. Analysis and optimal design of discrete order picking technologies along a line. *Naval Research Logistics (NRL)*, 55(4):350–362.

Ghiani G., **Laporte G., and Musmanno R. 2005**. *Introduction to Logistics Systems Planning and Control.* John Wiley & Sons, Ltd.

Gu J., **Goetschalckx M., and McGinnis L. 2007**. Research on warehouse operation: A comprehensive review. *European Journal of Operational Research*, 177(1):1–21.

Heizer J. and Render B. 2011. *Operations Management (Global edition)*. Pearson, 10th edition.

Kelly T. and Kupferschmid M. 1998. Numerical Verification of Second-Order Sufficiency Conditions for Nonlinear Programming. *SIAM-Review*, 40(2):310–314.

Kofler M., **Beham A.**, **Wagner S.**, **Affenzeller M., and Achleitner W. 2011**. Re-warehousing vs. healing: Strategies for warehouse storage location assignment. In *3rd IEEE International Symposium on Logistics and Industrial Informatics*, pages 77–82.

Koster R., **Le-Duc T., and Roodbergen K. 2006**. Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 182:481–501.

MathWorks® 2006. Null space of a sparse matrix. `http://de.mathworks.com/matlabcentral/fileexchange/11120-null-space-of-a-sparse-matrix`. Accessed: 27.06.2017 at 8 pm.

Nocedal J. and Wright S. 2006. *Numerical Optimization*. World Scientific, 2nd edition.

Petersen C. 1997. An evaluation of order picking routeing policies. *International Journal of Operations & Production Management*, 17(11):1098–1111.

Van den Berg J. 1999. A literature survey on planning and control of warehousing systems. *IIE Transactions*, 31(8):751–762.

List of Figures

List of Tables