



**HAL**  
open science

# Approximations based on the method of moving asymptotes

Abderrazak Driouch

► **To cite this version:**

Abderrazak Driouch. Approximations based on the method of moving asymptotes. Algebraic Geometry [math.AG]. Université de Pau et des Pays de l'Adour; Université Ibn Tofail. Faculté des sciences de Kénitra, 2020. English. NNT : 2020PAUU3043 . tel-03384750

**HAL Id: tel-03384750**

**<https://theses.hal.science/tel-03384750>**

Submitted on 19 Oct 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



PRÉSENTÉE POUR OBTENIR LE GRADE DE  
DOCTEUR DE L'UNIVERSITÉ IBN TOFAIL KENITRA  
ET  
DE L'UNIVERSITÉ DE PAU ET DES PAYS DE L'ADOUR

*Spécialité : Mathématiques appliquées*

par

**Abderrazak DRIOUCH**

**Approximations basées sur la méthode des asymptotes mobiles**  
**Approximations based on the method of moving asymptotes**

soutenue publiquement le 21 février 2020

Devant la commission d'examen composée de :

M. Bensouda Charaf	Professeur à l'Université Ibn Tofail Kenitra	Président
M. Domingo Barrera	Professeur à l'Université de Granada Espagne	Rapporteur
M. Dell'accio Francesco	Professeur à Université de Calabria Italie	Rapporteur
M. Achchab Boujema	Professeur à l'école Nationale des Sciences Appliquées (ENSA) Berrchid	Rapporteur
M. BECKER Roland	Professeur à l'Université de Pau et des Pays de l'Adour	Examinateur
M. Belmahjoub Fayçal	Professeur à l'Université Ibn Tofail Kenitra	Examinateur
M. GUESSAB Allal	Professeur à l'Université de Pau et des Pays de l'Adour	Directeur
M. Nouisser Otheman	Professeur à l'Université Ibn Tofail Kenitra	Directeur

LABORATOIRE DE MATHÉMATIQUES ET DE LEURS APPLICATIONS DE KENITRA (LMAK)  
ET LE LABORATOIRE DE MATHÉMATIQUES ET DE LEURS APPLICATIONS DE PAU (LMAP)



## Dedication

*To my dear father Elmaarouf Driouch,  
to express my great love and my deepest  
gratitude for his sacrifices and his great efforts,  
spent for my education and my well-being*

*To my dear mother Aziza Elamrani, there is no dedication quite  
eloquent to express my great love, my high regard and what  
you deserve for your prayers and your great sacrifices, which  
you have never ceased to provide me from my birth up to now*

*To my dear brother Abdelhak and my dear sisters Hafida, Latifa and  
Touria, to communicate my best feelings of fraternity, my high  
regard and my great love as recognition of your valuable  
tips, your encouragement and your huge support*

*To my whole family*

*To all my friends*

*To all my professors, from primary level to higher education,  
who taught and helped me to arrive here*

*I dedicate this modest work.*





## Acknowledgements

Acknowledgments Above all, I would like to thank the God almighty, my supreme creator for giving me the strength, the patience and the opportunity to achieve my best dream in the best possible conditions.

Firstly, I would like to address my sincere gratitude to my supervisors Prof. Allal Guessab and Prof. Otheman Nouisser for their support, their encouragements and their valuable guidance. I am very grateful for their valuable time spent in guiding my research and on sharing with me their expertise and their know-how. Finally, I would also like to say how much I appreciate their great availability and their professional and personal qualities.

My thanks must go also to my reporters Prof. ACHCHAB Boujemaa Prof. Dell'accio Francesco and Prof. Domingo Barrera for their precious time given for reading my dissertation and examine my works and also for establishing their reports. I am very thankful for their valuable comments which greatly improved the readability of my dissertation. I would also like to express my thanks to Prof. BENSOUA Charaf for giving me the honor of chairing my jury as well as Prof. BECKER Roland and Prof. Belmahjoub Fayçal, to accept to be a member of my dissertation committee.

I would also like to express my gratitude and appreciation to my origin laboratories, Laboratory of Mathematics and their Applications of Pau, and Laboratoy of Mathematics and their Applications of Kenitra, for the great working conditions, the numerous activities and the different training which it provides. I address my gratitude to all the teaching and administrative staff of the Faculty of Sciences of Kenitra, in particular, Prof. Benkirane Abdelmoujib for their discussions and their encouragements.

My acknowledgments go also to the director of the Laboratory of Mathematics and their Applications of Pau Prof. Gilles Carbou, and Jacques Giacomoni, for accepting me into his laboratory and for the great working conditions as well as for the numerous activities and training provided. My thanks also to all the teaching staff of the laboratory, in particular, Prof. Davide Trujillo for his time trying to fix with me a computer bug. I would like also to thank all the administrative staff of the laboratory, in particular Mr. Bruno Demoisy, Mrs. Sylvie Berton and Mrs. Chantal Blanchard.

My thanks and appreciation for the doctoral schools of exact sciences of Pau University and of sciences and techniques of Ibn Tofail University, which have significantly contributed in my scientific training through numerous training programs during my thesis.

I am grateful, too, to Prof. María Cruz López de Silanes and Enrique Artal for having welcomed me, for a research internship of six months, in his institute "Instituto de Matemáticas y estadísticas de la Universidad de Zaragoza". I am very thankful for him for giving me the opportunity to work in agreeable conditions and for his valuable time and help.

I would also like to thank the National Center of Scientific and Technical Research of Morocco, Campus France, and in particular the Laboratory of Mathematics and their Applications of Pau as well as the Erasmus+ program for the funding of my thesis.

I acknowledge all my professors, from the primary level up to now, who taught and educated me to arrive here, in particular, the teaching staff of Mathematics Department of Sciences and Techniques Faculties of Errachidia and of Fes. My special thanks go to my former professors in Fes, notably, Abdelmoujib Benkirane, Elhoussine Azroul, Jaouad BENNOUNA, Rachid Benkirane.

A very special thanks to all my family. Firstly, to my mother Aziza, my father Elmaarouf, my brother Abdelhak, my sisters Hafida, Latifa and Touria with their husbands Abdellah, Brahim and their children walid, mouad, Salma, Yahya and mohammed. Secondly, to all my uncles, aunts and cousins with their small families. I am very happy to have them by my sides and I am very grateful to each one of them.

Last but not least, I am grateful too, to all my lab mates and my friends PhD students at Ibn Tofail University and at Pau University. Firstly, for the former PhDs. I would like to thank my friends with whom I had a good time. Dears friends, your list is very long that I cannot cite you all, I wish you all happiness and full success in your personal and professional life.

---

## Résumé

La méthode des asymptotes mobiles (MMA) est largement utilisée pour minimiser une fonction continue  $f$  de plusieurs variables. À chaque itération de cette méthode, la fonction objective et les contraintes du problème d'optimisation sont approchés par une fonction rationnelle convexe. Pour assurer la convergence de la méthode MMA, le sous problème de chaque itération doit être résolu à son optimum global unique. Cette méthode formule de façon itérative des sous problèmes non linéaires séparables et strictement convexes. Des asymptotes inférieures et supérieures sont introduites pour tronquer la région réalisable. En raison de sa structure spéciale, les sous problèmes qui en résultent peuvent être résolus par de nombreuses méthodes efficaces d'optimisation non linéaire, par exemple les méthodes de points intérieurs (IPM) et la programmation séquentielle convexe (SCP).

La version originale de la méthode des asymptotes mobiles (MMA) n'est pas garantie à l'intérieur de la région réalisable correspondante décrite par les contraintes. Par conséquent, il n'est pas en mesure de résoudre les problèmes d'optimisation lorsque la région réalisable est définie par les contraintes de faisabilité.

Nous proposons dans cette thèse des nouvelles approximations et de nouveaux algorithmes d'optimisation sans et avec contraintes, faciles à mettre en œuvre sur la base de la méthode des asymptotes mobiles, ont les mêmes avantages que la version originale du MMA et du SCP, et plus d'avantages de convergence globale, et nous ne devons pas résoudre les sous problèmes générés par une autre méthode classique grâce à leur solution explicite. Pour montrer l'efficacité de notre algorithme, on les compare avec des méthodes connues comme la méthode de Newton.

Une extension de la MMA en utilisant les paramètres spectraux au lieu de l'information de second ordre est présentée, ces paramètres gardent la séquence générée commodément conservatrice par rapport aux fonctions originales et donnent une information sur la courbure, préservant la propriété de convergence globale. En ce qui concerne la fonction objective, des approximations conservatrices assurent des valeurs monotones décroissantes. La convexité stricte et la séparabilité des fonctions du modèle sont conservées afin que les sous problèmes générés aient une solution unique. Le but de l'utilisation de ces paramètres est de réduire l'effort total de calcul de l'algorithme et la possibilité de l'appliquer à des problèmes d'optimisation à grande échelle.

Dans une autre partie, nous proposons une modification de ces derniers algorithmes pour l'optimisation avec contraintes qui assure la faisabilité par rapport à un ensemble donné de contraintes inégalités. La procédure étend les sous problèmes résultants par des contraintes non linéaires supplémentaires, qui sont transmises directement au sous problème pour assurer leur faisabilité à chaque étape d'itération. Comme technique globale, une procédure de recherche de lignes est utilisée



pour assurer la convergence. Les sous problèmes qui en résultent peuvent être résolus efficacement en utilisant la structure clairsemée.

## Abstract

The method of moving asymptotes (MMA) is widely used for minimizing a continuous function  $f$  of several variables. At each iteration of this method, the objective function and the constraints of the optimization problem is approximated by a rational convex function. To ensure the convergence of the MMA method, the subproblem of each iteration needs to be solved to its unique global optimal. This method formulates separable and strictly convex nonlinear subproblems iteratively. Lower and upper asymptotes are introduced to truncate the feasible region. Due to the special structure, the resulting subproblems can be solved by numerous efficient nonlinear optimization methods, for example, interior-point methods (IPM) and sequential convex programming (SCP).

The original version of the Method of Moving Asymptotes (MMA) is not guaranteed to be in the corresponding feasible region described by the constraints. As a consequence, it is not able to solve the optimization problems where the feasible region defined by the constraints of feasibility.

We propose in this thesis a new approximations and algorithms for unconstrained optimization, easy to implement based on the method of moving asymptotes, have the same advantages of the original version of the MMA and the SCP, and more advantages of global convergence, and we do not need to solve the subproblems generated by other classical methods thanks to their explicit solutions. We compare the resulting algorithms with known methods like Newton's method.

An extension of the MMA using the spectral parameters instead of the second-order information is presented, these parameters keep the generated sequence conveniently conservative with respect to the original functions and give information about the curvature, preserving the global convergence property. As far as the objective function, conservative approximations ensure monotonically decreasing values. Strict convexity and separability of the model functions are kept so that the subproblems generated have a unique solution. The goal of using these parameters is to reduce the total effort computing of the algorithm and then the possibility to apply it for a large scale optimization problems.

In another part, we propose a modification of these algorithms for the constrained optimization that ensures feasibility with respect to a given set of inequality constraints. The procedure expands the resulting subproblems by additional nonlinear constraints, that are passed to the subproblem directly to ensure their feasibility in each iteration step. As globalization technique, a line search procedure is used to ensure convergence. The resulting subproblems can be solved efficiently taking the sparse structure.



# Contents

<b>List of Figures</b>	<b>xiii</b>
<b>List of Tables</b>	<b>xv</b>
<b>Introduction générale</b>	<b>3</b>
<b>1 Introduction</b>	<b>15</b>
<b>2 A GLOBALLY CONVERGENT MODIFIED UNIVARIATE VERSION OF THE METHOD OF MOVING ASYMPTOTES</b>	<b>27</b>
2.1 Motivation and theoretical justification . . . . .	27
2.2 A modified moving asymptotes method . . . . .	29
2.3 Convergence Analysis . . . . .	31
2.3.1 Convergence study . . . . .	34
2.3.2 Description of algorithm . . . . .	38
2.4 Numerical examples . . . . .	38
<b>3 A GLOBALLY CONVERGENT MULTIVARIATE VERSION OF THE MMA</b>	<b>43</b>
3.1 Motivation and theoretical justification . . . . .	43
3.2 A special modified multivariate version of moving asymptotes method . . . . .	46
3.3 Convergence Analysis . . . . .	48
3.3.1 Convergence study . . . . .	50
3.3.2 Choice of the parameters $\tilde{\alpha}^{(k)}$ . . . . .	54
3.3.3 Description of algorithm . . . . .	55
3.4 Minimizing Non-convex Non-Separable Functions . . . . .	55
3.5 Numerical examples . . . . .	57
3.6 Conclusion . . . . .	62
<b>4 NEW APPROACH OF MOVING ASYMPTOTES METHOD</b>	<b>63</b>
4.1 Introduction . . . . .	63
4.2 Univariate objective function . . . . .	64

4.2.1	Position of the problem and approximations . . . . .	64
4.2.2	Convergence analysis in $\mathbb{IR}$ . . . . .	69
4.3	The multivariate new approach of the MMA . . . . .	74
4.3.1	The multivariate setting . . . . .	75
4.4	Presentation of our method in $\mathbb{R}^{\times}$ . . . . .	76
4.4.1	Multivariate convergence result . . . . .	82
4.4.2	Rules of updating the spectral parameters $\eta^{(k)}$ , $d^{(k)}$ and $\sigma^{(k)}$ . . . . .	86
4.5	Numerical examples . . . . .	90
4.6	Conclusion . . . . .	98
<b>5</b>	<b>NEW SEQUENTIAL CONVEX PROGRAMMING METHOD BASED ON THE MMA</b>	<b>99</b>
5.1	Introduction . . . . .	99
5.2	Position of the problem . . . . .	100
5.3	Second order approximations . . . . .	104
5.4	Rules for updating the parameters $\rho_i^{(k,l)}$ , $\sigma_j^{(k)}$ and the asymptotes $d^{(k)}$ . . . . .	105
5.5	A New Sequential Convex Programming for constrained equality and/or inequality optimization . . . . .	109
5.5.1	A new algorithm based on the SCP method of Zillober . . . . .	109
5.6	A feasible sequential convex programming based on the NAMMA . . . . .	114
	<b>Bibliography</b>	<b>123</b>

# List of Figures

4.1	Graph of $f^{(k)}$ for different values of $\eta$ . . . . .	66
4.2	Graph of $f^{(k)}$ for different values of $d^{(k)}$ . . . . .	67
4.3	Behavior of the sequence $x^{(k)}$ . . . . .	70
4.4	Convergence $x^{(k)}$ , $X^{(k)}$ and $Y^{(k)}$ . . . . .	91
4.5	The evaluation's number of $\eta^{(k)}$ for our algorithm and SPG method . . . . .	95
5.1	Scheme of convergence iterations . . . . .	101



# List of Tables

2.1	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method. Here $w(x) = (1 +  x )^{1/2} \exp(-2 x )$ . . . . .	40
2.2	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present paper. Here $w(x) = (1 +  x )^{-4} \exp(-10 x ^{0.5}) * \log(e +  x )^{10}$ . . . . .	41
3.1	Numerical comparison of the [3] method, Newton's method, the BFGS Method and the present method. Here $w(x) = (1 +  x )^{1/2} \exp(-2 x )$ and $\tilde{\alpha}^{(k)} = 2 \left(1 + \frac{1}{4\tilde{c}^{(k)}}\right)$ . . . . .	58
3.2	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present paper. Here $w(x) = (1 +  x )^{-4} \exp(-10 x ^{0.5}) \log(e +  x )^{10}$ and $\tilde{\alpha}^{(k)} = 3 \left(1 + \frac{1}{10\tilde{c}^{(k)}}\right)$ . . . . .	59
3.3	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method. Here $w(x) = (1 +  x )^{1/2} \exp(-2 x )$ and $\tilde{\alpha}^{(k)} = \left(2 \left(1 + \frac{1}{4\gamma_1^{(k)}}\right), 4 \left(1 + \frac{1}{3\gamma_2^{(k)}}\right)\right)$ . 60	
3.4	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method. Here $w(x) = (1 +  x )^{1/2} \exp(-2 x )$ and $\tilde{\alpha}^{(k)} = \left(2 \left(1 + \frac{1}{5\gamma_1^{(k)}}\right), 3 \left(1 + \frac{1}{10\gamma_2^{(k)}}\right)\right)$ . 61	
3.5	Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method in three dimension. Here $w(x) = (1 +  x )^{1/4} \exp(-20 x )$ and $\tilde{\alpha}^{(k)} = \left(5 \left(1 + \frac{1}{7\gamma_1^{(k)}}\right), 2 \left(1 + \frac{1}{4\gamma_2^{(k)}}\right), 4 \left(1 + \frac{1}{3\gamma_3^{(k)}}\right)\right)$ . . . . .	61
3.6	Numerical result for non separable functions . . . . .	62
4.1	Variation table of $f$ . . . . .	72
4.2	Variation table of $f_i$ $i = 1, \dots, n$ . . . . .	84
4.3	Convergence of $x^{(k)}$ . . . . .	92
4.4	Numerical tests for non-convex non separable functions . . . . .	93







## List of symbols

$n$	number of primal variables ,
$x \in \mathbb{R}^n$	primal variable of dimension $n$ ,
$f(x)$	objective function ,
$m_e$	number of equality constraints ,
$c_j(x), j = m_e + 1, m_c$	number of equality and inequality constraints ,
$c_j(x), j = m_e + 1, \dots, m_c$	inequality constraints ,
$e_j(x), j = 1, \dots, m_f$	feasibility constraints,
$F$	feasible set given by the feasibility constraints $e_j(x)$ ,
$(k)$	iteration index,
$d^{(k)}$	moving asymptotes,
$\sigma^{(k)} \in (0, 1)$	stepsize to yield a descent in the merit function,
$y \in \mathbb{R}^{m_c+m_f}$	stepsize to yield a descent in the merit function,
$\mathbb{F}$	feasible region,
$\mathbb{J}(x)$	set of active constraints,
$L(x, y)$	Lagrangian function,
$y \in \mathbb{R}^{m_c+m_f}$	<i>dual variables</i> ,
$y_c \in \mathbb{R}^{m_c}$	dual variables with respect to $c_j(x) \quad j = 1, \dots, m_c$ ,
$y_e \in \mathbb{R}^{m_f}$	dual variables with respect to $e_j(x) \quad j = 1, \dots, m_f$ ,
<b>LICQ</b>	linear independence constraint qualification,
<b>KKT</b>	Karush-Kuhn-Tucker first order optimality conditions,
$z^{(k)} \in \mathbb{R}^n$	primal solution of the subproblem in iteration $k$ ,
$v^{(k)} \in \mathbb{R}^{m_c+m_f}$	dual solution of the subproblem in iteration $k$ ,
$r \in (0, 1)$	positive parameter used in Armijo condition,
$U_j \quad j = 1, \dots, n$	upper asymptote for primal variable $x_j \quad j = 1, \dots, n$ ,
$L_j \quad j = 1, \dots, n$	lower asymptote for primal variable $x_j \quad j = 1, \dots, n$ ,
$\mathbb{I}_+^{(k)}$	index set of nonnegative partial derivatives of objective,
$\mathbb{I}_-^{(k)}$	index set of negative partial derivatives of objective,
$f^{(k)}(x), \tilde{f}^{(k)}(x)$	approximations of the objective function $f$ in iteration $k$ ,
$\Phi_\rho(x, y)$	augmented Lagrangian merit function,
$\rho \in \mathbb{R}^{m_c+m_f}$	penalty parameters,
$\mathbb{J}^{(k)}$	set of active constraints with respect to augmented Lagrangian in iteration $k$ ,
$\eta_i^{(k)} \geq \eta^{(k)} > 0, i = 1, \dots, n$	estimation of curvature of $f^{(k)}(x)$ in variable $x_i; i = 1, \dots, n$ , in iteration $k$ .

# Introduction "version française"

La méthode des asymptotes mobiles (MMA) est introduite, sans une analyse globale de convergence, par Svanberg [69] en 1987. Cette méthode peut être considérée comme une généralisation de la méthode de linéarisation convexe (CONLIN); voire [11, 25, 26, 28, 48, 71, 73, 76, 80, 84]. Plus tard, Svanberg [72] à proposer une nouvelle version de la méthode qui converge globalement mais qui fait appel toujours à des méthodes classiques pour résoudre les sous-problèmes résultant des approximations de la méthode. Depuis lors, de nombreuses versions ont été suggérées.

Svanberg a proposé ce type spécial d'approximation convexe pour résoudre numériquement des problèmes d'optimisation structurels, ces méthodes d'optimisation structurelle doivent être flexible de résoudre non seulement les dimensions des éléments en tant que variables de conception,, mais aussi, par exemple, les variables de forme et les angles d'orientation des matériaux. Il devrait être capable de traiter toutes sortes de contraintes, à condition que seules les dérivées des fonctions de contraintes par rapport aux variables de conception puissent être calculées. Ainsi, la méthode devrait être capable de traiter des problèmes généraux de programmation non linéaire. En outre il doit tenir compte des caractéristiques des problèmes d'optimisation structurels, par exemple des évaluations de fonctions généralement très coûteuses, mais toujours la possibilité de calculer des gradients. De plus, la méthode devrait être stable et générer une séquence de solutions améliorées faisables du problème considéré. En effet, les approximations utilisées par Svanberg sont:

$$f_i^{(k)}(\mathbf{x}) = r_i^{(k)} + \sum_{j=1}^n \left( \frac{p_{ij}^{(k)}}{U_j^{(k)} - x_j} + \frac{q_{ij}^{(k)}}{x_j - L_j^{(k)}} \right) \quad (1)$$

où

$$p_{ij}^{(k)} = \begin{cases} (U_j^{(k)} - x_j^{(k)})^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j > 0 \\ 0, & \text{if } \partial f_i / \partial x_j \leq 0 \end{cases} \quad (2)$$

$$q_{ij}^{(k)} = \begin{cases} 0, & \text{if } \partial f_i / \partial x_j \geq 0 \\ -(x_j^{(k)} - L_j^{(k)})^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j < 0 \end{cases} \quad (3)$$

$$r_i^{(k)} = f_i(\mathbf{x}^{(k)}) - \sum_{j=1}^n \left( \frac{p_{ij}^{(k)}}{U_j^{(k)} - x_j^{(k)}} + \frac{q_{ij}^{(k)}}{x_j^{(k)} - L_j^{(k)}} \right) \quad (4)$$

où toutes les dérivées  $\partial f_i / \partial x_j$  pour  $i = 1, \dots, m$  et  $j = 1, \dots, n$  sont évaluées au  $x = x^{(k)}$ . Ainsi sous les conditions  $p_{ij}^{(k)} \geq 0$  et  $q_{ij}^{(k)} \geq 0$ ,  $f_i^{(k)}$  est une fonction convexe, et donc chaque sous-problème a un unique optimum global.

Par cette technique, la forme de chaque fonction d'approximation est spécifiée par deux valeurs sélectionnées de paramètres  $L_j^{(k)}$  et  $U_j^{(k)}$ , qui sont choisis en fonction de la MMA spécifique démarche. Plusieurs règles de sélection de ces valeurs sont expliquées en détail dans [14, 69]. Svanberg aussi montre comment ces deux paramètres peuvent être utilisés pour contrôler la convergence du processus. A chaque itération les asymptotes supérieure et inférieure  $U_j^{(k)}$  et  $L_j^{(k)}$  doivent être adaptées, les règles de mise à jour de ces paramètres selon Svanberg

- Si le processus a tendance à osciller, il doit être stabilisé. Cette stabilisation peut être réalisée en rapprochant les asymptotes du point d'itération courant.
- Si, au contraire, le processus est monotone et lent, il doit être " détendu ". Ceci peut être réalisé en éloignant les asymptotes du point d'itération courant.

Et donc une mise en oeuvre simple de ces règles est la suivante

Pour  $k = 0$  et  $k = 1$

$$L_j^{(k)} = x_j^{(k)} - (\bar{x}_j - x_j) \text{ et } U_j^{(k)} = x_j^{(k)} + (\bar{x}_j - \underline{x}_j). \quad (5)$$

Pour  $k \geq 2$

1. Si les signes de  $x_j^{(k)} - x_j^{(k-1)}$  et  $x_j^{(k-1)} - x_j^{(k-2)}$  sont opposées, indiquant une oscillation du processus itératif, et les valeurs de asymptotes sont données par

$$\begin{aligned} L_j^{(k)} &= x_j^{(k)} - s \left( x_j^{(k-1)} - L_j^{(k-1)} \right) \\ U_j^{(k)} &= x_j^{(k)} + s \left( U_j^{(k-1)} - x_j^{(k-1)} \right) \end{aligned} \quad (6)$$

2. Si les signes de  $x_j^{(k)} - x_j^{(k-1)}$  et  $x_j^{(k-1)} - x_j^{(k-2)}$  sont égaux, indiquant que le processus de convergence est lent, et donc

$$\begin{aligned} L_j^{(k)} &= x_j^{(k)} - \left( x_j^{(k-1)} - L_j^{(k-1)} \right) / s \\ U_j^{(k)} &= x_j^{(k)} + \left( U_j^{(k-1)} - x_j^{(k-1)} \right) / s \end{aligned} \quad (7)$$

Plusieurs règles heuristiques ont également été données pour un processus d'adaptation pour l'ajustement automatique de ces asymptotes à chaque itération; voire [69, 70] On peut résumer les caractéristiques les plus importantes de la MMA comme suit

- La MMA est une approximation de premier ordre au point itératif  $x^{(k)}$ , i.e.,

$$\begin{aligned} f^{(k)}(x^{(k)}) &= f(x^{(k)}) \\ \nabla f^{(k)}(x^{(k)}) &= \nabla f(x^{(k)}) \end{aligned} \quad (8)$$

- C'est une fonction rationnelle explicite, strictement convexe pour tout  $x$  tel que  $L_j^{(k)} < x_j < U_j^{(k)}$  avec pôles, et c'est monotone (croissante si  $\frac{\partial f}{\partial x_j}(x^{(k)}) > 0$  et décroissante si  $\frac{\partial f}{\partial x_j}(x^{(k)}) < 0$ ).
- Les approximations sont séparables, ce qui signifie que la fonction d'approximation  $F : \mathbb{R}^d \rightarrow \mathbb{R}$  peut être exprimé sous la forme d'une somme des fonctions des variables individuelles, c'est-à-dire, il existe des fonctions réelles  $F_1, F_2, \dots, F_d$  telles que

$$F(x) = F_1(x_1) + F_2(x_2) + \dots + F_d(x_d) \quad (9)$$

Une telle propriété est cruciale dans la pratique parce que les matrices Hessiennes des approximations seront diagonales, ce qui nous permet d'aborder des problèmes à grande échelle.

- Ce sont des fonctions lisses, les fonctions  $f^{(k)}$  sont deux fois continûment différentiable sur les intervalles  $L_j^{(k)} < x_j < U_j^{(k)}$ .
- A chaque itération externe, étant donné le point  $x^{(k)}$  actuel, un sous-problème est généré et résolu, et sa solution définit l'itération suivante  $x^{(k+1)}$ , donc une seule itération interne est effectuée.

Toutefois, il convient de mentionner que cette méthode ne donne pas de bons résultats dans certains cas. peut même échouer lorsque la courbure de l'approximation n'est pas correctement affectée [66]. En effet, il est important de comprendre que toutes les approximations convexes, y compris le MMA, qui sont fondées sur les valeurs des approximations de premier ordre, ne fournissent aucune information sur la courbure. La deuxième est contenue dans la matrice Hessienne de la fonction objective  $H_f$ , dont la composante  $(i, j)$  est  $\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$ . La mise à jour des asymptotes mobiles reste une tâche difficile. Une approche possible est d'utiliser la deuxième dérivée diagonale de la fonction objective pour définir les valeurs idéales de ces paramètres dans le MMA.

En fait, le MMA a été étendu afin d'inclure les dérivées de premier et de second ordre de la fonction objective. Un exemple simple de la MMA qui utilise un ordre de second ordre à l'itération

$x^{(k)}$  a été proposée par Fleury [27]

$$f^{(k)}(x) = f(x^{(k)}) + \sum_{j=1}^d \left( \frac{1}{x_j^{(k)} - a_j^{(k)}} - \frac{1}{x_j - a_j^{(k)}} \right) (x_j^{(k)} - a_j^{(k)})^2 \frac{\partial f}{\partial x_j}(x^{(k)}) \quad (10)$$

où, pour chaque  $j = 1, \dots, n$ , les asymptotes  $a_j^{(k)}$  sont déterminés en fonction des dérivées partielles du premier et de deuxième ordre par

$$a_j^{(k)} = x_j^{(k)} + 2 \frac{\frac{\partial f}{\partial x_j}(x^{(k)})}{\frac{\partial^2 f}{\partial x_j^2}(x^{(k)})} \quad (11)$$

et par [3]

$$\tilde{f}^{(k)}(x) = b^{(k)} + c^{(k)}(x - x^{(k)}) + d^{(k)} \left( \frac{1}{2} \frac{(x^{(k)} - a^{(k)})^3}{x - a^{(k)}} + \frac{1}{2} (x^{(k)} - a^{(k)}) (x - 2x^{(k)} + a^{(k)}) \right) \quad (12)$$

$$a^{(k)} = \begin{cases} L^{(k)} & \text{si } f'(x^{(k)}) < 0 \text{ et } L^{(k)} < x^{(k)} \\ U^{(k)} & \text{si } f'(x^{(k)}) > 0 \text{ et } U^{(k)} > x^{(k)} \end{cases} \quad (13)$$

Plusieurs versions ont été suggérées dans la littérature récente pour obtenir une mise en œuvre pratique du MMA qui tire pleinement parti de l'information de second ordre, par exemple [10, 17, 66], et les articles qui y sont cités fournissent des lectures supplémentaires sur ce sujet. Les limites de la méthode d'analyse asymptote pour les approximations convexes du premier ordre sont examinées par Smaoui et al [66], où l'on compare une approximation basée sur l'information du second ordre à une autre basée uniquement sur le premier ordre. L'approximation du second ordre permet d'obtenir le meilleur compromis entre robustesse et précision.

Contrairement à l'approche traditionnelle, nos méthodes remplacent le problème implicite (1.14) par une séquence de sous problèmes explicites convexes ayant une forme algébrique simple qui peut être résolue explicitement. Plus précisément, dans nos méthodes, une itération externe commence à partir de l'itération courante  $x^{(k)}$  et se termine par une nouvelle itération  $x^{(k+1)}$ . A chaque itération interne, à l'intérieur d'une itération externe explicite, un sous-problème convexe est généré et résolu. Dans ce sous-problème, la fonction objective originale est remplacée par une fonction linéaire plus une fonction rationnelle qui se rapproche des fonctions originales autour de  $x^{(k)}$ . La solution optimale du sous-problème devient  $x^{(k+1)}$ , et l'itération externe est terminée. Comme pour le MMA, nous montrerons que nos schémas d'approximation partagent toutes les caractéristiques énumérées ci-dessus.

De plus, notre méthode d'itération explicite est extrêmement simple à mettre en œuvre et facile à utiliser. De plus, le MMA est très pratique à utiliser dans la pratique, mais ses propriétés théoriques de convergence n'ont pas été étudiées de manière exhaustive. Le présent document présente une étude détaillée des propriétés de convergence de la méthode proposée.

La principale motivation de ce manuscrit était de proposer des schémas d'approximation qui, comme nous le verrons, répond à toutes les propriétés bien connues de convexité et de séparabilité de la MMA. En particulier, le régime que nous proposons présente les principaux avantages suivants :

1. Un aspect important de notre schéma d'approximation est que tous les sous-problèmes associés ont des solutions explicites.
2. Il génère une séquence d'itérations qui est limitée et converge vers un point fixe de la fonction objective.
3. Convergence globale de la suite générée par les sous-problèmes de la méthode.

Dans cette thèse, un ensemble d'algorithmes de programmation séquentiels convexe strictement réalisable est présenté. Le but est de générer une séquence d'itération qui est strictement réalisable pour une classe particulière de contraintes, appelées contraintes réalisables, tandis que d'autres contraintes peuvent être violées au cours du processus d'itération. L'algorithme est motivé par des résultats numériques sur des fonctions de Benchmark, où certaines contraintes et la fonction objective ne peuvent être évaluées que si certaines contraintes de faisabilité sont satisfaites. Autres caractéristiques typiques sont des racines carrées ou des fonctions logarithmes d'expressions analytiques. Nous procédons à partir de la formulation du problème suivante

$$\begin{aligned}
 \min_x \quad & f(x) \quad x \in \mathbb{R}^n \\
 \text{s.t.} \quad & c_j(x) = 0, \quad j = 1, \dots, m_e \\
 & c_j(x) \leq 0, \quad j = m_e + 1, \dots, m_c \\
 & e_j(x) \leq 0, \quad j = 1, \dots, m_f
 \end{aligned} \tag{14}$$

où les contraintes  $e_j(x), j = 1, \dots, m_f$ , sont au moins deux fois continûment différentiables sur  $\mathbb{R}^n$ . On suppose que certaines contraintes  $c_j(x), j = 1, \dots, m_e$ , et la fonction objective  $f(x)$  peuvent seulement être évalués sur l'ensemble faisable

$$F := \{x \in \mathbb{R}^n \mid e_j(x) \leq 0, j = 1, \dots, m_f\} \tag{15}$$

De plus, les contraintes régulières  $c_j(x), j = 1, \dots, m_c$  et la fonction objective  $f(x)$ , sont au moins deux fois différentiables et continues sur  $F$ . De plus, des contraintes de boîte peuvent être ajoutées au



problème d'optimisation (1.14), ce qui est généralement le cas dans la pratique.

Le développement de la méthode des asymptotes mobiles (MMA) est motivé par des résultats numériques et des comparaisons des algorithmes démontrés avec d'autres algorithmes bien connus, ces algorithmes sont appliqués en optimisation des matériaux libres et en optimisation topologique voir Bendsøe et al [6]. Ces méthodes peuvent considérer comme une extension des méthodes SCP et de la méthode des asymptotes mobiles, voir Bendsøe et Sigmund [64]. Dans un espace de conception donné, l'optimisation numérique trouve une approximation de l'optimum exacte du problème 1.14. La discrétisation par par élément fini est utilisée pour décider dans chaque élément s'il faut utiliser un matériau ou non. La rigidité de la structure est définie par la fonction dite de compliance, qui mesure le déplacement de la structure sous charge. Plus la conformité est faible, plus la structure résultante est rigide. De plus, la quantité totale de matériel est limitée. Pour éviter les instabilités numériques, c'est-à-dire les phénomènes en damier ou les zones grises, un filtre peut être utilisé, voir Ni, Zilber et Schittkowski[49]. Les problèmes d'optimisation en topologie sont des programmes non linéaires à grande échelle, qui peuvent être résolus efficacement par des algorithmes appropriés, par exemple la méthode de déplacement des asymptotes, voire Svanberg [69].

Les algorithmes proposés dans cette thèse sont une extension des méthodes de programmation convexe séquentielles (SCP), de programmation convexe strictement réalisable (SCPF), qui est fréquemment utilisée en génie mécanique, la première méthode ne garantit pas la faisabilité des itérations, c'est-à-dire  $m_f = 0$ , et la seconde nécessite une méthode par points intérieurs (IPM) pour obtenir une solution des sous problèmes produits. L'algorithme se rapproche de la solution optimale en résolvant une séquence de sous problèmes convexes et séparables, où une procédure de recherche de ligne par rapport à la fonction Lagrangienne augmentée est utilisée pour garantir la convergence globale, où la solution approximative satisfait toutes les contraintes du problème (1.14). MMA et SCP ont été conçus à l'origine pour résoudre des problèmes d'optimisation mécanique structurelle et sont souvent appliqués dans le domaine de l'optimisation topologique [63, 47, 13, 12]. Étant donné que dans certains cas particuliers, les contraintes structurelles typiques deviennent linéaires dans les variables inverses, une substitution appropriée est appliquée, qui devrait linéariser ces fonctions dans un certain sens, voire [86]. Les méthodes MMA sont dérivées de la méthode d'optimisation CONLIN (CONvex LINearization), voire [29, 24]. L'algorithme formule des sous-problèmes convexes et séparables en linéarisant les fonctions des problèmes par rapport à deux asymptotes flexibles, une inférieure et une supérieure, si la dérivée partielle est négative dans l'itération courante, alors nous utilisons l'asymptote inférieure. Sinon, il est linéarisé dans le sens original. Comme le succès du MMA et du SCP dépend du point de départ et de la méthode utilisée pour résoudre les sous problèmes générés, le processus peut se terminer en oscillation si l'un d'eux n'est pas bien choisi.

Les méthodes proposées incluent une nouvelle approximation de la fonction objective, et des contraintes et une procédure de recherche de ligne, car aucune preuve de convergence ne peut être

donnée pour la version originale du MMA. Les itérations sont évaluées par rapport à une fonction de mérite, qui combine la descente de la fonction objective et la faisabilité d'une manière appropriée. Le pas est réduit jusqu'à l'obtention d'une descente dans la fonction de mérite, par exemple, la fonction de Lagrange augmentée. Une stratégie active peut être appliquée pour réduire la taille du sous problème, économisant ainsi l'effort de calcul [5]. Le programme SCIP30.f est une implémentation de SCP, où la structure clairsemée des gradients et du Hessian est prise en compte. Quelques tests numériques comparatifs de SCP, de programmation quadratique séquentielle (SQP) et d'autres codes de programmation non linéaire sont disponibles pour des problèmes de tests d'optimisation structurels mécaniques, voire [62]. Une convergence globale de la méthode SCP est proposée par Zillober [82, 61].

Bien qu'aucune preuve de convergence ne puisse être donnée pour la version originale du MMA, l'algorithme donne de bons résultats dans la pratique. En 1995, [72] à présenter une extension qui est globalement convergente mais dans la plupart des cas pas aussi efficaces que le MMA d'origine. Par la suite, une nouvelle méthode de convergence globale appelée GCMMA a été mise au point, ce qui a donné de bons résultats dans la pratique. Elle ne s'applique qu'aux contraintes d'inégalité, c.-à-d.  $m_e = 0$ . Partant d'un point de départ réalisable  $x^{(0)} \in F$ , l'algorithme génère une séquence de points d'itération réalisables, c'est-à-dire  $m_c = 0$ ,  $m_f \neq 0$ . [72] à proposer des itérations internes supplémentaires garantissant

$$\begin{aligned} f\left(z^{(k,p)}\right) &\leq f^{(k,p)}\left(z^{(k,p)}\right) \\ e_j\left(z^{(k,p)}\right) &\leq e_j^{(k,p)}\left(z^{(k,p)}\right), \quad j = 1, \dots, m_f \end{aligned} \quad (16)$$

où  $f^{(k,p)}(x)$  est l'approximation strictement convexe de  $f(x)$  et  $e_j^{(k,p)}(x)$  est l'approximation convexe de  $e_j(x)$ ,  $j = 1, \dots, m_f$ , à l'iteration externe  $k$  et l'iteration interne  $p$ . De plus,  $z^{(k,p)} \in \mathbb{R}^n$  est la solution optimale du sous problème. Si la première ou la deuxième inégalité de (1.16) est violée pour au moins une contrainte ou la fonction objective, un sous problème plus conservateur est formulé à partir de l'approximation MMA. On peut montrer que la boucle d'itération interne se termine dans un nombre fini d'itérations. Notez que les fonctions doivent être évaluées à des points infaisables.

De nombreuses méthodes d'optimisation, par exemple SQP, appliquent des techniques de région de confiance pour montrer la convergence globale. Une nouvelle version du MMA est introduite par Ni [48], où les sous problèmes convexes sont limités par une région de confiance. Contrairement au MMA et au SCP, il ne s'applique qu'aux contraintes de boîte, alors que les contraintes d'égalité et d'inégalité ne peuvent être traitées, c'est-à-dire  $m_e = m_c = m_f = 0$ .

Une combinaison de la méthode de déplacement des asymptotes avec l'approche par filtre est proposée par Fletcher et Leyffer [21], est donnée par Ertel [18]. Un itéré est accepté, si une descente de la fonction objective ou une réduction de la violation des contraintes est obtenue. Sinon, le point

est rejeté et un nouveau sous problème est généré en réduisant la distance entre les asymptotes. Les méthodes de filtrage induisent une séquence d'itération non monotone. Pour plus d'informations sur la méthode du filtre SQP et la preuve de convergence de l'algorithme, voire [22].

Une extension de la méthode SCP proposée par [68] pour les programmes semi-définis appelés PENSQP. Ils considèrent le problème suivant

$$\begin{aligned}
 \min_z \quad & f(Z) \quad Z \in \mathbb{S}^n \\
 \text{s.t.} \quad & c_j(Z) \leq 0, \quad j = 1, \dots, m_c \\
 & Z - \underline{Z} \succeq 0 \\
 & \bar{Z} - Z \succeq 0
 \end{aligned} \tag{17}$$

où  $\mathbb{S}^n$  désigne l'espace des matrices symétriques de taille  $n$ . L'algorithme crée une séquence d'approximations convexes séparables par blocs du premier ordre. Contrairement au MMA et au SCP, la méthode utilise des asymptotes constantes. De plus, une procédure de recherche de ligne est appliquée pour assurer une descente suffisante de la fonction objective. Le sous problème semi-défini qui en résulte peut-être résolu efficacement grâce à sa structure spécifique par des solutions appropriées, par exemple PENNON, voire [67]. La convergence globale de l'algorithme résultant peut être montrée, voire [68]. Comme GCMMA on obtient de bons résultats pour les problèmes d'optimisation de topologie, il doit être appliqué à l'optimisation des matériaux libres [65, 56, 42]. Certaines fonctions propres aux problèmes de FMO ne sont définies qu'à l'intérieur de la région réalisable, compte tenu des contraintes de faisabilité  $e_j(x)$ ,  $j = 1, \dots, m_f$ . La méthode GCMMA proposée par Svanberg [73] n'assure pas la faisabilité des itérations pendant le processus de solution. Par conséquent, nous devons étendre la méthode de telle sorte que la stricte faisabilité soumise à un ensemble de contraintes particulières soit garantie à chaque étape de l'itération. Les contraintes de faisabilité convexes sont transmises directement au sous-problème tandis que la fonction objective ainsi que les contraintes restantes sont approximées sur la base du schéma d'approximation proposé dans [36, 37]. Une stratégie d'ensemble actif n'est appliquée que pour les contraintes restantes, afin d'assurer la faisabilité chaque fois que des fonctions ou des gradients doivent être évaluées. En outre, les contraintes que l'on s'attend à trouver dans la solution optimale sont toujours incluses dans le réglage actif de la structure clairsemée des gradients et de la matrice Hessienne. Une recherche de ligne est effectuée pour assurer la convergence globale.

Les méthodes d'optimisation réalisables calculent une séquence d'itérations réalisables, c'est-à-dire que seules les contraintes de faisabilité  $e_j(x)$ ,  $j = 1, \dots, m_f$ , sont considérées, et donc  $m_e = m_c = 0$ . Dans la littérature, plusieurs méthodes d'optimisation réalisables peuvent être trouvées. Dans de nombreuses applications du monde réel, les problèmes d'optimisation sont de grande envergure et l'évaluation du gradient de fonction peut prendre du temps. En utilisant des techniques d'optimisation réalisables, le processus d'optimisation peut être interrompu à chaque itération, ce qui donne une solution réalisable, mais non optimale. Les méthodes d'optimisation réalisables les plus connues sont

les méthodes de points intérieurs réalisables, les méthodes de projection et les méthodes de direction réalisables.

En général, les méthodes de points intérieurs (IPM) calculent à chaque itération une direction de descente de Newton en résolvant un système linéaire d'équations. L'orientation de la recherche qui en résultera pourrait ne pas être réalisable. Par conséquent, un deuxième système linéaire est formulé où le côté droit est perturbé assurant une direction réalisable. Certaines des méthodes FDIP résolvent un troisième système linéaire pour assurer une convergence super linéaire près d'un point fixe. Analogue aux méthodes SQP réalisables, une recherche de ligne le long de l'arc de recherche est effectuée pour assurer à la fois la faisabilité et une descente de la fonction objective. Un certain nombre de méthodes réalisables de point intérieur sont données dans la littérature [4, 38, 39, 51, 81, 52].

Les méthodes de point intérieur réalisable partent de l'intérieur de la région réalisable et calculent une séquence d'itération qui s'approche de la limite. Une sous-classe est une méthode de barrière, où un paramètre de barrière combine les contraintes et la fonction objectif. Cela revient à ce qu'on appelle la fonction de barrière qui doit être minimisée, par exemple, par la méthode de Newton. Généralement, la fonction de barrière n'est définie que sur la région réalisable et tend vers l'infini à la limite. Une fonction de barrière populaire est la fonction de barrière logarithmique

$$f(x) + \mu \sum_{i=1}^{m_f} \ln(-e_j(x)) \quad (18)$$

où  $\mu \in \mathbb{R}^+$  est le paramètre barrière. En commençant par un gros  $\mu$ , il est réduit itérativement de telle sorte que des solutions proches de la frontière peuvent être obtenues. Ces méthodes sont particulièrement efficaces pour les problèmes d'optimisation convexe, voire [54, 33, 53, 1]. Une autre classe de méthodes d'optimisation réalisables sont les méthodes de projection. Dans chaque itération  $k$ , les algorithmes calculent une direction de recherche  $d^{(k)} \in \mathbb{R}^n$  et projeter le point résultant  $x^{(k)} + d^{(k)}$  à la limite de la région réalisable, si nécessaire. Le point projeté sur la frontière est noté  $x_p^{(k)} \in \mathbb{R}^n$ . La direction de recherche projetée  $d_p^{(k)} \in \mathbb{R}^n$  se compose de deux composants. À l'intérieur de la région réalisable, la direction de recherche projetée est donnée par  $d^{(k)}$ . La deuxième partie est décrite par le segment de la limite entre le point d'intersection de  $d^{(k)}$  avec la limite et le point de projection  $x_p^{(k)}$ . Une recherche de ligne est effectuée le long de la direction de recherche projetée  $d_p^{(k)}$ . Pour assurer la faisabilité, les problèmes doivent être convexes. L'effort de calcul de la projection dépend de l'algorithme et des contraintes du problème d'optimisation. Quelques méthodes de projection populaires sont présentées par Rosen [58, 57] et par Grippo and al [35]. Les méthodes de projection sont souvent combinées avec d'autres méthodes efficaces d'optimisation non linéaire pour calculer la direction de descente  $d^{(k)}$ . Jian, Zhang et Xue [41] ont combiné une méthode SQP avec des méthodes de projection pour obtenir une méthode SQP réalisable. Le sous problème quadratique est résolu pour obtenir une direction de descente. De plus, par une projection de l'itération  $x^{(k)}$  sur la frontière on

obtient un nouvel itéré.

Les méthodes de direction réalisables calculent une direction faisable  $d^{(k)}$ , ce qui assure l'existence de  $\theta^{(k)} \in \mathbb{R}^+$ , tel que  $x^{(k)} + \sigma^{(k)}d^{(k)}$  est faisable pour tout  $\sigma^{(k)} \leq \theta^{(k)}$ , où  $\sigma^{(k)} \in \mathbb{R}$  est la taille du pas. De nombreuses méthodes d'orientation réalisables peuvent être trouvées dans la littérature [87]. Le premier algorithme de direction réalisable est proposé par Zoutendijk en 1960, voire [75]. Dans chaque itération  $k$ , une direction de recherche réalisable améliorée est déterminée et une recherche de ligne étendue est effectuée, donnant une descente suffisante de la fonction objectif et satisfaisant les contraintes  $e_j(x) \leq 0$ ,  $j = 1, \dots, m_f$ . Commençons par un point de conception faisable  $x^{(0)}$ , et calculons une direction de recherche  $d^{(k)}$  à l'itération  $k$  satisfait une direction de descente par rapport à la fonction objectif et les  $\varepsilon$ -contraintes actives  $\mathbb{J}_\varepsilon^{(k)} := \{j = 1, \dots, m_f | e_j(x^{(k)}) \geq -\varepsilon\}$ ,  $\varepsilon \in \mathbb{R}^+$ , i.e.,

$$\begin{aligned} \nabla f(x^{(k)})^T d^{(k)} &\leq 0 \\ \nabla e_j(x^{(k)})^T d^{(k)} &\leq 0, \quad j \in \mathbb{J}_\varepsilon^{(k)} \end{aligned} \quad (19)$$

A chaque itération, un sous-problème linéaire est formulé, qui maximise la descente minimale, voir [74, 44]. Nous dénotons la solution de

$$\begin{aligned} \min_{\delta, d} \quad & \delta \quad d \in \mathbb{R}^n, \delta \in \mathbb{R} \\ \text{s.t.} \quad & \nabla f(x^{(k)})^T d \leq \delta \\ & \nabla e_j(x^{(k)})^T d \leq \delta, \quad j \in \mathbb{J}_\varepsilon^{(k)} \\ & \|d\|_\infty \leq 1 \end{aligned} \quad (20)$$

par  $(d^{(k)}, \delta^{(k)})$ . Si  $\varepsilon$  est adapté correctement, on peut montrer que  $\delta^{(k)} \leq 0$ , pour tout  $k = 0, 1, \dots$ . La taille de  $\varepsilon$  est très importante pour la convergence de l'algorithme. Si  $\varepsilon$  devient trop petit, on observe un comportement oscillant typique de l'algorithme. En 1961, Zoutendijk [75], a développé un algorithme plus robuste que le premier, et  $\varepsilon$  n'a pas besoin d'être adapté.

En résolvant le problème

$$\begin{aligned} \min_{\delta, d} \quad & \delta \quad d \in \mathbb{R}^n, \delta \in \mathbb{R} \\ \text{s.t.} \quad & \nabla f(x^{(k)})^T d \leq \delta \\ & e_j(x^{(k)}) + \nabla e_j(x^{(k)})^T d \leq \delta, \quad j \in \mathbb{J}_\varepsilon^{(k)}, \\ & \|d\|_\infty \leq 1 \end{aligned} \quad (21)$$

on obtient une solution  $(d^{(k)}, \delta^{(k)})$ , avec  $\mathbb{J}_\varepsilon^{(k)} := \{j = 1, \dots, m_f | e_j(x^{(k)}) \geq -\varepsilon\}$ . Une preuve de convergence pour les deux méthodes peut être donnée pour les contraintes convexes  $e_j$ ,  $j = 1, \dots, m_f$ , voire [7]. Pour ces deux dernières versions des méthodes de direction de recherche, seule la convergence

linéaire peut être représentée. Par conséquent, les sous problèmes sont étendus de sorte que les informations du deuxième ordre sont incluses. Une possibilité est de calculer une direction de descente  $d_0^{(k)} \in \mathbb{R}^n$  en résolvant un sous problème quadratique (QP), c'est-à-dire une fonction objective quadratique et des contraintes linéaires, selon les méthodes SQP, voir Schittkowski et Yuan [30]

$$\begin{aligned} \min_d \quad & \frac{1}{2} d^T H(x^{(k)}, y^{(k)}) d + \nabla f(x^{(k)})^T d \quad d \in \mathbb{R}^n \\ \text{s.t.} \quad & e_j(x^{(k)}) + \nabla e_j(x^{(k)})^T d \leq 0, \quad j = 1, \dots, m_f \end{aligned} \quad (22)$$

où  $H(x^{(k)}, y^{(k)}) \in \mathbb{R}^{n \times n}$  est la matrice hessienne de la fonction lagrangienne par rapport à  $x$  ou une approximation appropriée. De plus  $y^{(k)} \in \mathbb{R}; m_f$  est la variable double. La direction de recherche résultante  $d_0^{(k)}$  peut ne pas être réalisable, comme pour les contraintes actives  $\nabla e_j(x^{(k)})^T d_0^{(k)} = 0$  est autorisée, ce qui donne une direction tangentielle à la région réalisable [50]. Par conséquent, une correction est déterminée en inclinant la direction d'origine vers la région réalisable. Pour assurer une convergence rapide à proximité d'une solution, une direction de recherche supplémentaire est calculée par pliage. Une recherche de ligne étendue est effectuée le long de l'arc de recherche composé des trois directions, de sorte que la faisabilité et une descente suffisante de la fonction objectif sont garanties. La complexité de calcul par itération des méthodes SQP réalisables est significativement plus élevée par rapport aux méthodes SQP habituelles. Dans les méthodes les plus modernes, la complexité des calculs a été réduite.

Cette thèse traite des approximations basées sur la méthode des asymptotes mobiles. L'objectif principal est de proposer une nouvelle approche et des algorithmes faciles à mettre en œuvre pour résoudre un problème d'optimisation non linéaire non convexe sans et avec un type particulier de contraintes et de comparer les résultats avec des méthodes connues. Au chapitre 2 publié dans le journal "Applicable Analysis and Discrete Mathematics", [37], présente le premier travail de cette thèse, une nouvelle méthode modifiée d'asymptotes mobiles est présentée. À chaque étape du processus itératif, un sous problème approximatif strictement convexe est généré et explicitement résolu, et ce faisant, nous proposons une stratégie pour incorporer une information de second ordre modifiée pour la localisation des asymptotes en mouvement. Cela réduit considérablement le coût de calcul de notre méthode d'optimisation et peut à la fois stabiliser et accélérer la convergence du processus général. Sous des hypothèses naturelles, nous prouvons la convergence géométrique de l'algorithme d'optimisation associé. De plus, les résultats expérimentaux révèlent que la méthode actuelle est significativement plus rapide que la méthode [3], la méthode de Newton et la méthode BFGS, et qu'elle réussira lorsque ces dernières divergent simultanément.

Au chapitre 3, publié dans Applicable Analysis and Discrete Mathematics [36], nous présentons une extension de notre article précédent dans un cadre multivarié. La version multivariée proposée est un résultat globalement convergent pour une nouvelle méthode, qui consiste itérativement en la

solution d'une version modifiée de la méthode de déplacement des asymptotes. Il est montré que l'algorithme généré possède certaines propriétés souhaitables. Nous indiquons les conditions dans lesquelles la convergence géométrique de la méthode actuelle est garantie. Tous nos résultats expérimentaux montrent que l'algorithme obtenu est significativement plus rapide que la méthode [3], la méthode de Newton et la méthode CG-BFGS, et il réussira là où ces dernières divergent simultanément.

Dans le chapitre 4, nous présentons une nouvelle approche de la Méthode de Déplacement des Asymptotes utilisant des paramètres de projection spectrale pour l'optimisation par contrainte liée, ce nouvel algorithme combine les deux avantages de la convexité et de la séparabilité du MMA et l'accélération de la convergence de la méthode du gradient projeté. L'avantage de ce nouvel algorithme est qu'il est très facile à mettre en œuvre et très efficace pour les problèmes à grande échelle et garder la faisabilité de l'itérer.

Au chapitre 5, une extension de la nouvelle approche de la méthode de déplacement des asymptotes présentée au chapitre 4 pour les problèmes d'optimisation contraints est proposée, basée sur les méthodes de programmation séquentielle convexe proposées par [86]. L'idée est d'inclure les informations de second ordre fournies par ce paramètre dans les fonctions du modèle qui définissent les approximations rationnelles de la fonction objectif et les contraintes non linéaires, aux dépens d'évaluations de gradient supplémentaires par itération interne. Le paramètre spectral permet de conserver la séquence générée de façon pratique et conservatrice par rapport aux fonctions d'origine, tout en préservant la propriété de convergence globale. En ce qui concerne la fonction objectif, des approximations conservatrices assurent des valeurs décroissantes monotones, alors que pour les contraintes, la faisabilité d'un problème est garantie. La convexité stricte et la séparabilité du modèle sont conservées afin que les sous-problèmes aient une solution unique.

Enfin, le chapitre 6 donne les conclusions de cette thèse et quelques suggestions pour les travaux futurs.

# Chapter 1

## Introduction

The method of moving asymptotes (MMA) was introduced, without a global convergence analysis, by Svanberg [69] in 1987. This method can be considered as a generalization of the convex linearization method (CONLIN); see [11, 25, 26, 28, 48, 71–73, 76, 80, 84], Later, Svanberg [72] proposed a new version of the method that converges globally but still uses classical methods to solve the subproblems resulting from method approximations. Since then, many versions have been suggested.

Svanberg proposed this special type of convex approximation to solve structural optimization problems numerically, these structural optimization methods must be flexible to solve not only the dimensions of the elements as design variables, but also, for example, shape variables and material orientation angles. It should be able to handle all kinds of constraints, provided that only derivatives of the constraint functions in relation to the design variables can be calculated. Thus, the method should be able to deal with general nonlinear programming problems. In addition, it must take into account the characteristics of structural optimization problems, for example, evaluations of functions that are generally very expensive, but always the possibility of calculating gradients. In addition, the method should be stable and generate a sequence of feasible, improved solutions to the problem under consideration. Indeed, the approximations used by Svanberg are:

$$f_i^{(k)}(\mathbf{x}) = r_i^{(k)} + \sum_{j=1}^n \left( \frac{p_{ij}^{(k)}}{U_j^{(k)} - x_j} + \frac{q_{ij}^{(k)}}{x_j - L_j^{(k)}} \right) \quad (1.1)$$

where

$$p_{ij}^{(k)} = \begin{cases} (U_j^{(k)} - x_j^{(k)})^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j > 0 \\ 0, & \text{if } \partial f_i / \partial x_j \leq 0 \end{cases} \quad (1.2)$$

$$q_{ij}^{(k)} = \begin{cases} 0, & \text{if } \partial f_i / \partial x_j \geq 0 \\ -(x_j^{(k)} - L_j^{(k)})^2 \partial f_i / \partial x_j, & \text{if } \partial f_i / \partial x_j < 0 \end{cases} \quad (1.3)$$



$$r_i^{(k)} = f_i(\mathbf{x}^{(k)}) - \sum_{j=1}^n \left( \frac{p_{ij}^{(k)}}{U_j^{(k)} - x_j^{(k)}} + \frac{q_{ij}^{(k)}}{x_j^{(k)} - L_j^{(k)}} \right) \quad (1.4)$$

where all derivatives  $\partial f_i / \partial x_j$  for  $i = 1, \dots, m$  and  $j = 1, \dots, n$  are valued at  $x = x^{(k)}$ . Thus under the conditions  $p_{ij}^{(k)} \geq 0$  and  $q_{ij}^{(k)} \geq 0$ ,  $f_i^{(k)}$  is a convex function, and therefore each subproblem has a unique global optimum.

By this technique, the shape of each approximation function is specified by two selected parameter values  $L_j^{(k)}$  and  $U_j^{(k)}$ , which are chosen according to the specific MMA approach. Several rules for selecting these values are explained in detail in [14, 69]. Svanberg also shows how these two parameters can be used to control process convergence. At each iteration the upper and lower asymptotes  $U_j^{(k)}$  and  $L_j^{(k)}$  must be adapted, the rules for updating these parameters according to Svanberg

- If the process tends to oscillate, it must be stabilized. This stabilization can be achieved by bringing the asymptotes closer to the current iteration point.
- If, on the other hand, the process is monotonous and slow, it must be "relaxed". This can be achieved by moving the asymptotes away from the current iteration point.

And so a simple implementation of these rules is as follows

for  $k = 0$  and  $k = 1$

$$L_j^{(k)} = x_j^{(k)} - (\bar{x}_j - x_j) \text{ and } U_j^{(k)} = x_j^{(k)} + (\bar{x}_j - x_j). \quad (1.5)$$

For  $k \geq 2$

1. If the signs of  $x_j^{(k)} - x_j^{(k-1)}$  and  $x_j^{(k-1)} - x_j^{(k-2)}$  are opposite, indicating an oscillation of the iterative process, and the asymptomatic values are given by

$$\begin{aligned} L_j^{(k)} &= x_j^{(k)} - s \left( x_j^{(k-1)} - L_j^{(k-1)} \right) \\ U_j^{(k)} &= x_j^{(k)} + s \left( U_j^{(k-1)} - x_j^{(k-1)} \right) \end{aligned} \quad (1.6)$$

2. If the signs of  $x_j^{(k)} - x_j^{(k-1)}$  and  $x_j^{(k-1)} - x_j^{(k-2)}$  are equal, indicating that the convergence process is slow, and therefore

$$\begin{aligned} L_j^{(k)} &= x_j^{(k)} - \left( x_j^{(k-1)} - L_j^{(k-1)} \right) / s \\ U_j^{(k)} &= x_j^{(k)} + \left( U_j^{(k-1)} - x_j^{(k-1)} \right) / s \end{aligned} \quad (1.7)$$

Several heuristic rules have also been given for an adaptation process for the automatic adjustment of these asymptotes to each iteration; see [69, 70] The most important characteristics of MMA can be summarized as follows

- The MMA is a first-order approximation at the iterative point  $x^{(k)}$ , i.e.,

$$\begin{aligned} f^{(k)}(x^{(k)}) &= f(x^{(k)}) \\ \nabla f^{(k)}(x^{(k)}) &= \nabla f(x^{(k)}) \end{aligned} \quad (1.8)$$

- It is an explicit rational function, strictly convex for all  $x$  such that  $L_j^{(k)} < x_j < U_j^{(k)}$  with poles, and it's monotonous. (increasing if  $\frac{\partial f}{\partial x_j}(x^{(k)}) > 0$  and decreasing if  $\frac{\partial f}{\partial x_j}(x^{(k)}) < 0$ ).
- The approximations are separable, which means that the approximation function  $F : \mathbb{R}^d \rightarrow \mathbb{R}$  can be expressed as a sum of the functions of the individual variables, i. e. there are real functions  $F_1, F_2, \dots, F_d$  such that

$$F(x) = F_1(x_1) + F_2(x_2) + \dots + F_d(x_d) \quad (1.9)$$

Such a property is crucial in practice because the Hessian matrices of the approximations will be diagonal, allowing us to address problems on a large scale.

- These are smooth functions, the functions  $f^{(k)}$  are twice continuously differentiable over the intervals  $L_j^{(k)} < x_j < U_j^{(k)}$ .
- For each external iteration, given the current point  $x^{(k)}$ , a subproblem is generated and solved, and its solution defines the next iteration  $x^{(k+1)}$ , so only one internal iteration is performed.

However, it should be mentioned that this method does not give good results in some cases. May even fails when the curvature of the approximation is not correctly affected [66]. Indeed, it is important to understand that all convex approximations, including MMA, which are based on the values of first order approximations, do not provide any information on curvature. The second is contained in the Hessian matrix of the objective function  $H_f$ , whose component  $(i, j)$  is  $\frac{\partial^2 f}{\partial x_i \partial x_j}(x)$ . Updating moving asymptotes remains a difficult task. One possible approach is to use the second diagonal derivative of the objective to define the ideal values of these parameters in the MMA. In fact, the MMA has been extended to include first- and second-order derivatives of the objective function. For example, a simple example of MMA using a second order at iteration  $x^{(k)}$  was proposed by Fleury [27]

$$\begin{aligned} f^{(k)}(x) &= f(x^{(k)}) \\ &+ \sum_{j=1}^d \left( \frac{1}{x_j^{(k)} - a_j^{(k)}} - \frac{1}{x_j - a_j^{(k)}} \right) (x_j^{(k)} - a_j^{(k)})^2 \frac{\partial f}{\partial x_j}(x^{(k)}) \end{aligned} \quad (1.10)$$

, where, for each  $j = 1, \dots, n$ , the asymptotes  $a_j^{(k)}$  are determined as a function of the first and second order partial derivatives by

$$a_j^{(k)} = x_j^{(k)} + 2 \frac{\frac{\partial f}{\partial x_j}(x^{(k)})}{\frac{\partial^2 f}{\partial x_j^2}(x^{(k)})} \quad (1.11)$$

, and by [3]

$$\begin{aligned} \tilde{f}^{(k)}(x) = & b^{(k)} + c^{(k)}(x - x^{(k)}) \\ & + d^{(k)} \left( \frac{1}{2} \frac{(x^{(k)} - a^{(k)})^3}{x - a^{(k)}} + \frac{1}{2} (x^{(k)} - a^{(k)}) (x - 2x^{(k)} + a^{(k)}) \right) \end{aligned} \quad (1.12)$$

$$a^{(k)} = \begin{cases} L^{(k)} & \text{if } f'(x^{(k)}) < 0 \text{ and } L^{(k)} < x^{(k)} \\ U^{(k)} & \text{if } f'(x^{(k)}) > 0 \text{ and } U^{(k)} > x^{(k)} \end{cases} \quad (1.13)$$

Several versions have been suggested in the recent literature to achieve a practical implementation of the MMA that takes full advantage of second-order information, for example, [10, 17, 66], and the articles cited therein to provide additional reading on this subject. The limitations of the asymptote analysis method for first-order convex approximations are examined by Smaoui et al [66], where an approximation based on second-order information is compared to an approximation based only on the first order. The second-order approximation makes it possible to obtain the best compromise between robustness and precision.

Unlike the traditional approach, our methods replace the implicit problem (4.1) with a sequence of explicit convex subproblems with a simple algebraic form that can be solved explicitly. More precisely, in our methods, an external iteration starts from the current iteration  $x^{(k)}$  and ends with a new iteration  $x^{(k+1)}$ . At each internal iteration, within an explicit external iteration, a convex sub-problem is generated and solved. In this sub-problem, the original objective function is replaced by a linear function plus a rational function that approximates the original functions around  $x^{(k)}$ . The optimal solution of the subproblem becomes  $x^{(k+1)}$ , and the external iteration is complete. As with MMA, we will show that our approximation schemes share all the characteristics listed above. In addition, our explicit iteration method is extremely simple to implement and easy to use. In addition, MMA is very practical to use in practice, but its theoretical properties of convergence have not been studied in an exhaustive way. This document presents a detailed study of the convergence properties of the proposed method.

The main purpose of this manuscript was to propose approximation schemes which, as we will see, meet all the well-known properties of convexity and separability of MMA. In particular, the regime we propose has the following main advantages:

1. An important aspect of our approximation scheme is that all associated sub-problems have explicit solutions.
2. It generates an iteration sequence that is limited and converges to a fixed point of the objective function.
3. Global convergence of the sequence generated by the method's subproblems.

In this thesis, a set of strictly feasible convex sequential programming algorithms is presented. The goal is to generate an iteration sequence that is strictly feasible for a particular class of constraints, called feasible constraints, while other constraints may be violated during the iteration process. The algorithm is motivated by numerical results on Benchmark functions, where certain constraints, and the objective function can only be evaluated if certain feasibility constraints are met. Other typical characteristics are square roots or logarithmic functions of analytical expressions. We proceed from the following problem formulation

$$\begin{aligned}
\min_x \quad & f(x) \quad x \in \mathbb{R}^n \\
\text{s.t.} \quad & c_j(x) = 0, \quad j = 1, \dots, m_e \\
& c_j(x) \leq 0, \quad j = m_e + 1, \dots, m_c \\
& e_j(x) \leq 0, \quad j = 1, \dots, m_f
\end{aligned} \tag{1.14}$$

where the constraints  $e_j(x), j = 1, \dots, m_f$ , are at least twice continuously differentiable on  $\mathbb{R}^n$ . It is assumed that some constraints  $c_j(x), j = 1, \dots, m_e$ , and the objective function  $f(x)$  can only be evaluated on the feasible set

$$F := \{x \in \mathbb{R}^n \mid e_j(x) \leq 0, j = 1, \dots, m_f\} \tag{1.15}$$

In addition, the regular constraints  $c_j(x), j = 1, \dots, m_c$  and the objective function  $f(x)$ , are at least twice continuously differentiable on  $F$ . In addition, box constraints can be added to the optimization problem 1, which is usually the case in practice.

The development of the Method of Moving Asymptotes(MMA) is motivated by numerical results and comparisons of demonstrated algorithms with other well-known algorithms, these algorithms are applied in free material optimization and topological optimization [6]. These methods can be considered as an extension of the SCP methods and the Method of Moving Asymptotes, see [64]. In a given design space, numerical optimization finds an approximation of the exact optimum of the problem 1.14. Finite element discretization is used to decide in each element whether or not to use a material. The rigidity of the structure is defined by the so-called compliance function, which measures the displacement of the structure under load. The lower the compliance, the more rigid the resulting structure. In addition, the total amount of material is limited. To avoid numerical instabilities, i. e. checkered phenomena or grey areas, a filter can be used, see Ni, Zillober and Schittkowski [49].

Topology optimization problems are large-scale, non-linear programs that can be solved effectively, by appropriate algorithms, such as the method of moving asymptotes, see [69].

The algorithms proposed in this thesis are also an extension of the sequential convex programming (SCP), the strict feasible convex programming (SCPF) methods, which is frequently used in mechanical engineering, the first method does not ensure the feasibility of the iterates, i.e.,  $m_f = 0$ , and the second need an interior point method (IPM) to get a solution of the subproblems generated. The algorithm approximates the optimal solution by solving a sequence of convex and separable subproblems, where a line search procedure with respect to the augmented Lagrangian merit function is used for guaranteeing global convergence, where the approximate solution satisfies all the constraints of the problem (1.14). MMA and SCP were originally designed for solving structural mechanical optimization problems and it is often applied in the field of topology optimization [63, 47, 13, 12]. Due to the fact that in some special cases, typical structural constraints become linear in the inverse variables, a suitable substitution is applied, which is expected to linearize these functions in some sense, see [86] MMA methods are derived from the optimization method CONLIN (CONvex LINearization), see [29, 24]. The algorithm formulates convex and separable subproblems by linearizing the problem's functions with respect to two flexible asymptotes, a lower and an upper one, if the partial derivative is negative in the current iterate, then we use the lower asymptote. Otherwise, it is linearized in the original sense. As the success of MMA and SCP is dependent on the starting point and the method used to solve the sub-problems generated, and the process might end in oscillation if one of them is not well chosen.

The methods proposed, including a new approximations of the objective function and constraints, and a line search procedure, as no convergence proof can be given to the original version of MMA. The iterates are valuated with respect to a merit function, which combines the descent of the objective function and the feasibility in a suitable way. The stepsize is reduced until a descent in the merit function, e.g, the augmented Lagrangian function, is obtained. An active strategy can be applied to reduce the size of the subproblem, saving computational effort [5]. The program SCPIP30.f is an implementation of SCP, where the sparse structure of the gradients and the Hessian is taken into account. Some comparative numerical tests of SCP, sequential quadratic programming (SQP) and some other nonlinear programming codes are available for tests problems from mechanical structural optimization, see [62]. Zillober in [61, 82] show the global convergence of the SCP.

Although no convergence proof for the original version of MMA can be given, the algorithm yields good results in practice. In 1995, [72] presented an extension which is globally convergent but in most cases not as efficient as the original MMA version. Later on, a new globally convergent method called GCMMA (globally convergent method of moving asymptotes) was developed, yielding good results in practice. It is only applicable for inequality constraints, i.e.,  $m_e = 0$ . Proceeding from a feasible starting point  $x^{(0)} \in F$ , the algorithm creates a sequence of feasible iterates, i.e.,  $m_c = 0$ ,

$m_f \neq 0$ . [72] proposed additional inner iterations ensuring

$$\begin{aligned} f(z^{(k,p)}) &\leq f^{(k,p)}(z^{(k,p)}) \\ e_j(z^{(k,p)}) &\leq e_j^{(k,p)}(z^{(k,p)}), \quad j = 1, \dots, m_f \end{aligned} \quad (1.16)$$

where  $f^{(k,p)}(x)$  is the strictly convex approximation of  $f(x)$  and  $e_j^{(k,p)}(x)$  is the convex approximation of  $e_j(x)$ ,  $j = 1, \dots, m_f$ , in outer iteration  $k$  and the inner iteration  $p$ . Moreover,  $z^{(k,p)} \in \mathbb{R}^n$  is the optimal solution of the subproblem. If the first or the second inequality of (1.16) is violated for at least one constraint or the objective function, a more conservative subproblem is formulated based on the MMA approximation. It can be shown that the inner iteration loop terminates within a finite number of iterations. Note that the functions have to be evaluated at infeasible points.

Many optimization methods, for example SQP, apply trust region techniques to show global convergence. A new version of MMA is introduced by Ni [48], where the convex subproblems are restricted by a trust region. In contrast to MMA and SCP, it is only applicable for the box constraints, while equality and inequality constraints cannot be handled, i.e.,  $m_e = m_c = m_f = 0$ .

A combination of the method of moving asymptotes with the filter approach proposed by Fletcher and Leyffer [21], is given by Ertel [18]. An iteration is accepted, if a descent in the objective function or a reduction of the constraint violation is obtained. Otherwise, the point is rejected and a new subproblem is generated by reducing the distance between the asymptotes. Filter methods induce a non-monotone iteration sequence. For more information about the SQP-filter method and the proof of convergence of the algorithm, see [22]. An extension of the SCP method proposed by [68] for semidefinite programs called PENSCP. They consider the following problem

$$\begin{aligned} \min_z \quad & f(Z) \quad Z \in \mathbb{S}^n \\ \text{s.t.} \quad & c_j(Z) \leq 0, \quad j = 1, \dots, m_c \\ & Z - \underline{Z} \succeq 0 \\ & \bar{Z} - Z \succeq 0 \end{aligned} \quad (1.17)$$

where  $\mathbb{S}^n$  denotes the space of symmetric matrices of size  $n$ . The algorithm creates a sequence of first-order block-separable convex approximations. In contrast to MMA and SCP, the method uses constant asymptotes. Moreover, a line search procedure is applied to ensure a sufficient descent in the objective function. The resulting semidefinite subproblem can be solved efficiently due to its specific structure by appropriate solvers, e.g., PENNON, see [67]. Global convergence of the resulting algorithm can be shown, see [68]. As GCMMA achieves good results for topology optimization problems, it is to be applied to free material optimization [42, 56, 65]. Some of the problem specific functions of FMO are only defined within the feasible region given by feasibility constraints  $e_j(x)$ ,  $j = 1, \dots, m_f$ . The GCMMA method proposed by Svanberg [73] is not ensuring feasibility of the

iterates during the solution process. Therefore, we need to extend the method such that strict feasibility subject to a special set of constraints is guaranteed in each iteration step. The convex feasibility constraints are passed to the subproblem directly while the objective function, as well as the remaining constraints are approximated based on the approximation's scheme proposed in [36, 37]. An active set strategy is applied to the remaining constraints only, to ensure feasibility whenever functions or gradients are to be evaluated. In addition, constraints that are expected to be active in the optimal solution are always included in the active setting the sparse structure of the gradients and Hessian. A line search is performed to ensure global convergence.

Feasible optimization methods compute a sequence of feasible iterates, i.e., only feasibility constraints  $e_j(x)$ ,  $j = 1, \dots, m_f$ , are considered, i.e.,  $m_e = m_c = 0$ . In the literature, several feasible optimization methods can be found. In many real-world applications, the optimization problems are of large scale dimension and the function gradient evaluations might be time-consuming. Using feasible optimization techniques, the optimization process can be aborted at each iterate yielding a feasible, although not optimal solution. The most known feasible optimization methods are feasible interior-point methods, projection methods, and feasible direction methods.

In general, interior-point methods (IPM) compute in each iteration a Newton descent direction by solving a linear system of equations. The resulting search direction might not be a feasible direction. Therefore, a second linear system is formulated where the right-hand side is perturbed ensuring a feasible direction. Some of the FDIP methods solve a third linear system to ensure superlinear convergence near a stationary point. Analog to feasible SQP methods, a line search along the search arc is performed to ensure both feasibility and a descent in the objective function. Several feasible direction interior methods are given in the literature [4, 38, 39, 51, 81, 52].

Feasible interior point methods start from the interior of the feasible region and compute an iteration sequence that approaches the boundary. A subclass is a barrier method, where a barrier parameter combines the constraints and the objective function. This yields the so-called barrier function which is to be minimized, e.g., by Newton's method. Typically the barrier function is only defined on the feasible region and tends to infinity at the boundary. A popular barrier function is the logarithmic barrier function

$$f(x) + \mu \sum_{i=1}^{m_f} \ln(-e_j(x)) \quad (1.18)$$

where  $\mu \in \mathbb{R}^+$  is the barrier parameter. Starting with a large  $\mu$ , it is reduced iteratively, such that solutions near the boundary can be obtained. These methods are especially successful for convex optimization problems, see [54, 33, 53, 1].

Another class of feasible optimization methods is projection methods. In each iteration  $k$ , the algorithms compute a search direction  $d^{(k)} \in \mathbb{R}^n$  and project the resulting point  $x^{(k)} + d^{(k)}$  on the

boundary of the feasible region, if necessary. The projected point on the boundary is denoted  $x_p^{(k)} \in \mathbb{R}^n$ . The projected search direction  $d_p^{(k)} \in \mathbb{R}^n$  consists of two components. Inside the interior of the feasible region, the projected search direction is given by  $d^{(k)}$ . The second part is described by the segment of the boundary between the intersection point of  $d^{(k)}$  with the boundary and the projection point  $x_p^{(k)}$ . A line search is performed along the projected search direction  $d_p^{(k)}$ . To ensure feasibility, the problems have to be convex.

The effort to compute the projection depends on the algorithm and on the constraints of the optimization problem. Some popular projection methods are presented by Rosen [58, 57], and by Grippo and al [35]. Projection methods are often combined with other efficient nonlinear optimization methods to compute the descent direction  $d^{(k)}$ . Jian, Zhang and Xue [41] combined an SQP method with projection methods to get feasible SQP methods. The quadratic subproblem is solved to obtain a decent direction. Moreover, By a projection of the iterate  $x^{(k)}$  on the boundary we get a new iteration.

Feasible direction methods compute a feasible direction  $d^{(k)}$ , which ensures the existence of  $\theta^{(k)} \in \mathbb{R}^+$ , such that  $x^{(k)} + \sigma^{(k)}d^{(k)}$  is feasible for all  $\sigma^{(k)} \leq \theta^{(k)}$ , where  $\sigma^{(k)} \in \mathbb{R}$  is the stepsize. Many different feasible direction methods can be found in the literature [87]. The first feasible direction algorithm is proposed by Zoutendijk in 1960, see [75]. In each iteration  $k$ , an improving feasible search direction is determined and an extended line search is performed, yielding a sufficient descent in the objective function and satisfying the constraints  $e_j(x) \leq 0$ ,  $j = 1, \dots, m_f$ . Let us start from a feasible design point  $x^{(0)}$ , and compute a search direction  $d^{(k)}$  at the iteration  $k$  satisfies a descent direction with respect to the objective function and the  $\varepsilon$  active constraints  $\mathbb{J}_\varepsilon^{(k)} := \{j = 1, \dots, m_f | e_j(x^{(k)}) \geq -\varepsilon\}$ ,  $\varepsilon \in \mathbb{R}^+$ , i.e.,

$$\begin{aligned} \nabla f(x^{(k)})^T d^{(k)} &\leq 0 \\ \nabla e_j(x^{(k)})^T d^{(k)} &\leq 0, \quad j \in \mathbb{J}_\varepsilon^{(k)} \end{aligned} \quad (1.19)$$

At each iteration, a linear subproblem is formulated, which maximizes the minimal descent, see [74, 44]. We denote the solution of

$$\begin{aligned} \min_{\delta, d} \quad & \delta \quad d \in \mathbb{R}^n, \delta \in \mathbb{R} \\ \text{s.t.} \quad & \nabla f(x^{(k)})^T d \leq \delta \\ & \nabla e_j(x^{(k)})^T d \leq \delta, \quad j \in \mathbb{J}_\varepsilon^{(k)} \\ & \|d\|_\infty \leq 1 \end{aligned} \quad (1.20)$$

by  $(d^{(k)}, \delta^{(k)})$ . If  $\varepsilon$  is adapted adequately, it can be shown that  $\delta^{(k)} \leq 0$ , for all  $k = 0, 1, \dots$ . The size of  $\varepsilon$  is very important for the convergence of the algorithm. If  $\varepsilon$  becomes too small, we observe a



typical oscillating behavior of the algorithm. In 1961, Zoutendijk [75], developed an algorithm more robust than the first, and  $\varepsilon$  need not be adapted.

By solving the problem

$$\begin{aligned} \min_{\delta, d} \quad & \delta \quad d \in \mathbb{R}^n, \delta \in \mathbb{R} \\ \text{s.t.} \quad & \nabla f(x^{(k)})^T d \leq \delta \\ & e_j(x^{(k)}) + \nabla e_j(x^{(k)})^T d \leq \delta, \quad j \in \mathbb{J}_\varepsilon^{(k)}, \\ & \|d\|_\infty \leq 1 \end{aligned} \quad (1.21)$$

we get a solution  $(d^{(k)}, \delta^{(k)})$ , with  $\mathbb{J}_\varepsilon^{(k)} := \{j = 1, \dots, m_f | e_j(x^{(k)}) \geq -\varepsilon\}$ . A convergence proof for both methods can be given for convex constraints  $e_j$ ,  $j = 1, \dots, m_f$ , see [7]. For these two last versions of the search direction methods, only linear convergence can be shown. Therefore, the subproblems are extended such that second-order information is included. One possibility is to compute a descent direction  $d_0^{(k)} \in \mathbb{R}^n$  by solving a quadratic subproblem (QP), i.e., a quadratic objective function and linear constraints, according to SQP methods, see [30]

$$\begin{aligned} \min_d \quad & \frac{1}{2} d^T H(x^{(k)}, y^{(k)}) d + \nabla f(x^{(k)})^T d \quad d \in \mathbb{R}^n \\ \text{s.t.} \quad & e_j(x^{(k)}) + \nabla e_j(x^{(k)})^T d \leq 0, \quad j = 1, \dots, m_f \end{aligned} \quad (1.22)$$

where  $H(x^{(k)}, y^{(k)}) \in \mathbb{R}^{n \times n}$  is the Hessian of the Lagrangian function with respect to  $x$  or an appropriate approximation. Moreover  $y^{(k)} \in \mathbb{R}^{m_f}$  is the dual variable. The resulting search direction  $d_0^{(k)}$  may not be feasible, as for active constraints  $\nabla e_j(x^{(k)})^T d_0^{(k)} = 0$  is allowed, which yields to a search direction tangential to the feasible region [50]. Therefore, a correction is determined by tilting the original direction towards the feasible region. To ensure fast convergence near a solution an additional search direction is computed by bending. An extended line search is performed along the search arc consisting of all three directions, such that the feasibility and a sufficient descent in the objective function are guaranteed. The computational complexity per iteration of the feasible SQP methods is significantly higher compared to the usual SQP methods. In state-of-the-art methods, the computational complexity has been reduced.

This thesis deals with approximations based on the method of moving asymptotes. The main objective is to propose a new approach and algorithms easy to implement for solving a non-convex nonlinear optimization problem without and with a special kind of constraint and compare the results with a known method.

In Chapter 2 published in "Applicable Analysis and Discrete Mathematics" [37], presents the first work during my thesis, A new modified moving asymptotes method is presented. At each step of the iterative process, a strictly convex approximating subproblem is generated and explicitly solved, and

in doing so, we propose a strategy to incorporate modified second-order information for the moving asymptotes location. This considerably reduces the computational cost of our optimization method and may both stabilize and speed up the convergence of the general process. Under natural assumptions, we prove the geometric convergence of the associated optimization algorithm. In addition, experimental results reveal that the present method is significantly faster compared to the [3] method, Newton's method, and the BFGS Method, and it will succeed where this latter diverges simultaneously.

In Chapter 3, published in *Applicable Analysis and Discrete Mathematics* [36], we introduce an extension of our previous paper [37] in a multivariate setting. The proposed multivariate version is a globally convergent result of a new method, which consists iteratively of the solution of a modified version of the method of moving asymptotes. It is shown that the algorithm generated has some desirable properties. We state the conditions under which the present method is guaranteed to converge geometrically. All our experimental results show that the resulting algorithm is significantly faster compared to the [3] method, Newton's method, and the CG-BFGS Method, and it will succeed where this latter diverge simultaneously.

In Chapter 4, we present a new approach of the Method of Moving Asymptotes using a spectral projected parameter for bound constrained optimization, this new algorithm combines both advantages of convexity and separability of MMA and the speed up the convergence of the projected gradient method. The advantage of this new algorithm is that it is quite easy to implement and very effective for large scale problems and keep the feasibility of the iteration.

In Chapter 5, an extension of the new approach of the method of moving asymptotes presented in chapter 4 for constrained optimization problems is proposed, based on the sequential convex programming methods as proposed by [86]. The idea is to include the second-order information provided by this parameter into the model functions that define the rational approximations to the objective function and the nonlinear constraints, at the expense of additional gradient evaluations per inner iteration. The spectral parameter keeps the generated sequence conveniently conservative with respect to the original functions, preserving the global convergence property. As far as the objective function, conservative approximations ensure monotonically decreasing values, whereas, for the constraints, feasibility for an augmented problem is guaranteed. Strict convexity and separability of the model are kept so that the subproblems have a unique solution. Finally, Chapter 6 gives the conclusions of this thesis and some suggestions for future works.



## Chapter 2

# A GLOBALLY CONVERGENT MODIFIED UNIVARIATE VERSION OF THE METHOD OF MOVING ASYMPTOTES

### 2.1 Motivation and theoretical justification

Consider the unconstrained optimization problem: Find  $x_* \in \Omega$  such that

$$f(x_*) = \min_{x \in \Omega} f(x), \quad (2.1)$$

where  $\Omega$  is an open subset of  $\mathbb{R}$  and  $f : \mathbb{R} \rightarrow \mathbb{R}$  is a given non-linear real-valued objective function, typically twice continuously differentiable, which could be non-convex. In order to evaluate the merit of using second order information an extension of the method of moving asymptotes, that accounts for the curvatures, was proposed in [3]. Let us first briefly recall its main idea. Throughout, we assume that  $f'$  does not vanish at a given suitable initial point  $x^{(0)} \in \Omega$ , that is  $f'(x^{(0)}) \neq 0$ , since if this is not the case we have nothing to solve. Starting from the initial design point  $x^{(0)}$  the iterates  $x^{(k)}$  are computed successively by solving sub-problems of the form: Find  $x^{(k+1)}$  such that

$$f^{(k)}(x^{(k+1)}) = \min_{x \in \Omega} f^{(k)}(x), \quad (2.2)$$

where the approximating function  $f^{(k)}$  of the objective function  $f$  at the  $k$ -th iteration has the following form:

$$f^{(k)}(x) = a^{(k)} + b^{(k)}(x - x^{(k)}) + c^{(k)} \left( \frac{1}{2} \frac{(x^{(k)} - d^{(k)})^3}{x - d^{(k)}} + \frac{1}{2} (x^{(k)} - d^{(k)})(x - 2x^{(k)} + d^{(k)}) \right), \quad (2.3)$$

with

$$d^{(k)} = \begin{cases} L^{(k)} & \text{if } f'(x^{(k)}) < 0, \text{ and } L^{(k)} < x^{(k)} \\ U^{(k)} & \text{if } f'(x^{(k)}) > 0, \text{ and } U^{(k)} > x^{(k)}, \end{cases} \quad (2.4)$$

here, the asymptotes  $U^{(k)}$  and  $L^{(k)}$  are adjusted heuristically during the iterations, or guided by a proposed given function where the first and second derivative are evaluated at the current iteration point  $x^{(k)}$ . In contrast with the classical Newton method, here the approximation functions  $f^{(k)}$  are of the form of a *linear* function plus a *rational* function. For each iteration, the approximate parameters  $a^{(k)}$ ,  $b^{(k)}$  and  $c^{(k)}$  used in equation (2.3) are determined in such a way that the following set of interpolation conditions are satisfied:

$$f^{(k)}(x^{(k)}) = f(x^{(k)}), \quad (2.5)$$

$$(f^{(k)})'(x^{(k)}) = f'(x^{(k)}), \quad (2.6)$$

$$(f^{(k)})''(x^{(k)}) = f''(x^{(k)}). \quad (2.7)$$

It follows from the above identities that  $a^{(k)}$ ,  $b^{(k)}$ , and  $c^{(k)}$  are given by

$$\begin{aligned} a^{(k)} &= f(x^{(k)}), \\ b^{(k)} &= f'(x^{(k)}), \\ c^{(k)} &= f''(x^{(k)}). \end{aligned} \quad (2.8)$$

In order to apply this method, it is necessary that the objective function should fulfill for each iteration  $k$  the following condition:

$$f''(x^{(k)}) > 0, \quad (2.9)$$

which is typically an important weakness to this approach. Hence, this method is very restrictive and also has the following disadvantage :

- It needs good initial solution  $x_*$  close to the exact solution.
- It converges slowly, in many cases, to the optimum  $x_*$ .
- It does not always converge.
- Its performance degrades when it applied to nonconvex functions.

It is the intention of this contribution to extend the [3] method to a more general context, by removing the restrictive condition (2.9) on the objective functions making the present approach more efficient. It is shown, as will be proved below, that the new method converges geometrically. Comparative numerical studies also show the success of the proposed extensions for various kinds of different test functions. Moreover, in almost all the problems we consider, the method of the present paper works better than the [3] method, Newton's method or the BFGS method itself.

This chapter is organized as follows. In Section 2.2, we discuss a modified moving asymptotes method. The general convergence theory for the new moving asymptotes method. In section 2.4, various numerical experiments conduct to confirm our theoretical finding.

## 2.2 A modified moving asymptotes method

Throughout this chapter we assume that  $w$  is a function satisfying the following conditions:

$$w \text{ is a real-valued function, defined and continuous on } \mathbb{R}, \quad (2.10)$$

$$\lim_{|x| \rightarrow +\infty} w(x) = 0. \quad (2.11)$$

Our general modification of moving asymptotes method that we examine herein may be described as follows: Given the iteration point  $\tilde{x}^{(k)}$  (at iteration  $k$ ).

- The objective function  $f$  is iteratively approximated at the  $k$ -th iteration by the approximating function  $\tilde{f}_w^{(k)}$  where:

$$\tilde{f}_w^{(k)}(x) = \tilde{a}^{(k)} + \tilde{b}^{(k)}(x - \tilde{x}^{(k)}) + \tilde{c}^{(k)} \left( \frac{1}{2} \frac{(\tilde{x}^{(k)} - \tilde{d}^{(k)})^3}{x - \tilde{d}^{(k)}} + \frac{1}{2} (\tilde{x}^{(k)} - \tilde{d}^{(k)})(x - 2\tilde{x}^{(k)} + \tilde{d}^{(k)}) \right). \quad (2.12)$$

- The approximating function  $\tilde{f}_w^{(k)}$  is first order approximations of the original function  $f$  at the current iteration point  $\tilde{x}^{(k)}$ , i.e.,

$$\tilde{f}_w^{(k)}(\tilde{x}^{(k)}) = f(\tilde{x}^{(k)}), \quad (2.13)$$

$$(\tilde{f}_w^{(k)})'(\tilde{x}^{(k)}) = f'(\tilde{x}^{(k)}). \quad (2.14)$$

In addition to the above conditions (2.13) and (2.14), the approximating function should satisfy the more general condition (2.15) instead of (2.7):

$$(\tilde{f}_w^{(k)})''(\tilde{x}^{(k)}) = \left| f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)}) \right|. \quad (2.15)$$

Consequently, in the present situation, the approximate parameters  $\tilde{a}^{(k)}$ ,  $\tilde{b}^{(k)}$  and  $\tilde{c}^{(k)}$  are here determined for each iteration such that:

$$\tilde{a}^{(k)} = f(\tilde{x}^{(k)}), \quad (2.16)$$

$$\tilde{b}^{(k)} = f'(\tilde{x}^{(k)}), \quad (2.17)$$

$$\tilde{c}^{(k)} = \left| f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)}) \right|. \quad (2.18)$$

Furthermore, in order to fully determine an explicit expression for the approximating function  $\tilde{f}_w^{(k)}$ , the parameter  $\tilde{d}^{(k)}$  is chosen such that

$$\tilde{d}^{(k)} = \tilde{x}^{(k)} + 2\tilde{\alpha}^{(k)} \frac{f'(\tilde{x}^{(k)})}{\tilde{c}^{(k)}}, \quad (2.19)$$

where  $\{\tilde{\alpha}^{(k)}\}_k$  is a sequence of real numbers with

$$\tilde{\alpha}^{(k)} > 1, (k \in \mathbb{N}). \quad (2.20)$$

Different rules for how to choose these values (and possible weight functions in (2.15)) will be discussed later. We note that our method does not use the interpolation condition (2.7), but instead we have incorporated a first- and second-order information, as given in (2.18). Moreover, in particular, if you take  $w = 0$  and at each iteration condition (2.9) is fulfilled, then our iterative scheme obviously reduces to the one introduced in [3]. Hence, subsequent iterations of the [3] method are similar, except that in the proposed approximating function  $\tilde{f}_w^{(k)}$  the parameters  $c^{(k)}$  and  $d^{(k)}$  are replaced by those computed in (2.18) and (2.19) respectively. It starts at an initial point  $\tilde{x}^{(0)}$  and generates successive iterates by

$$f(\tilde{x}^{(k+1)}) \leftarrow \tilde{f}_w^{(k)}(\tilde{x}^{(k+1)}) = \min_{x \in \Omega} \tilde{f}_w^{(k)}(x). \quad (2.21)$$

For simplicity, we have removed the index  $w$  in  $\tilde{x}_w^{(k)}$ .

We prefer to work with (2.18) instead of (2.7) for several reasons. First, as mentioned above, this allows us to to apply our method to a large class of objective functions. There is also a significant difference from a numerical point of view: many experimental results reveal that the iterative scheme based on our modification (2.18) can yield significantly fewer iterations than the [3] method, Newton's method or the BFGS Method itself. In contrast to these three approaches, our method converges even if the starting point is very far from the true solution. In addition, as we will see, the key features of the present method are:

- It does not require us to build a good initial solution close to the exact solution.
- It converges geometrically for a large class of functions  $w$  that satisfy condition (2.11).
- It will succeed where the [3] method, Newton's method and the BFGS Method break down.

We have not succeeded in proving that the method can be extended to multiple dimensions, but in practice, we have found it to work in two dimensions.

### 2.3 Convergence Analysis

We start this section with a result concerning an explicit expression for the iterative sequence  $\{\tilde{x}^{(k)}\}_k$  generated by the approximating function  $\tilde{f}_w^{(k)}$ . Here, we continue to denote by  $\tilde{c}^{(k)}$ ,  $\tilde{d}^{(k)}$  and  $\tilde{\alpha}^{(k)}$  the coefficients given by (2.18), (2.19) and (2.20) respectively. For brevity, in the following we use the notation:

$$\mathcal{I}_k = \begin{cases} ]\tilde{d}^{(k)}, +\infty[ & \text{if } f'(\tilde{x}^{(k)}) < 0, \\ ]-\infty, \tilde{d}^{(k)}[ & \text{if } f'(\tilde{x}^{(k)}) > 0. \end{cases}, \quad (2.22)$$

**Theorem 2.3.1** *With the above notation, let  $\Omega \subset \mathbb{R}$  be an open subset of the real line, a given twice continuously differentiable function  $f$  in  $\Omega$ ,  $\tilde{x}^{(0)} \in \Omega$  and  $\tilde{x}^{(k)}$  being respectively the initial and a current point of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$ . Then, for each  $k > 0$  the approximating function defined by (2.12) is a strictly convex function on  $\mathcal{I}_k$ . In addition, the function  $\tilde{f}_w^{(k)}$  has a unique minimum at*

$$\tilde{x}^{(k+1)} \leftarrow \tilde{x}_*^{(k)} = \tilde{d}^{(k)} + (\tilde{x}^{(k)} - \tilde{d}^{(k)})\sqrt{\tilde{s}^{(k)}} \quad (2.23)$$

where

$$\tilde{s}^{(k)} = \frac{\tilde{\alpha}^{(k)}}{\tilde{\alpha}^{(k)} - 1}. \quad (2.24)$$

**Proof 1** *The proof for this theorem follows by arguments essentially identical to those given in [3]. The main ingredient here is a suitable application of the condition (2.20) imposed on the coefficient  $\tilde{d}^{(k)}$ . Next, let us make the following observation. We have tacitly assumed that the denominator of (2.24) cannot have the value zero, since if it is the case then by (2.19) and (2.20) we immediately get  $f'(\tilde{x}^{(k)}) = 0$ , and we have nothing to show. We first start by showing that the approximating function  $\tilde{f}_w^{(k)}$  is strictly convex in  $\mathcal{I}_k$ . To this end we prove that  $(\tilde{f}_w^{(k)})''$  is nonnegative in  $\mathcal{I}_k$ . Indeed, a simple calculation reveals that*

$$\left(\tilde{f}_w^{(k)}\right)''(x) = \tilde{c}^{(k)} \left(\frac{\tilde{x}^{(k)} - \tilde{d}^{(k)}}{x - \tilde{d}^{(k)}}\right)^3. \quad (2.25)$$

*In view of (2.25), it remains to show that the term on the right-hand side of (2.25) is nonnegative for all  $x \in \mathcal{I}_k$ . Since  $\tilde{c}^{(k)}$  is nonnegative (as can be seen in (2.18)), and the two terms  $x_k - \tilde{d}^{(k)}$  and  $x - \tilde{d}^{(k)}$  have the same sign in the interval  $\mathcal{I}_k$ , then  $\tilde{f}_w^{(k)}$  is a convex function on  $\mathcal{I}_k$ . Furthermore, the function  $\tilde{f}_w^{(k)}$  being continuous in  $\mathcal{I}_k$ , this implies the existence of a minimum, which by convexity is the unique critical point  $\tilde{x}_*^{(k)}$ . Now, looking for  $(\tilde{f}_w^{(k)})'(x) = 0$  we conclude that the optimum  $\tilde{x}_*^{(k)}$  is*



one solution of the equation

$$f'(\tilde{x}^{(k)}) + \frac{1}{2}\tilde{c}^{(k)}(\tilde{x}^{(k)} - \tilde{d}^{(k)}) \left(1 - \frac{(\tilde{x}^{(k)} - \tilde{d}^{(k)})^2}{(x - \tilde{d}^{(k)})^2}\right) = 0, \quad (2.26)$$

which, after trivial calculations, implies

$$\left(\frac{\tilde{x}^{(k)} - \tilde{d}^{(k)}}{x - \tilde{d}^{(k)}}\right)^2 = 1 + \frac{2f'(\tilde{x}^{(k)})}{\tilde{c}^{(k)}(\tilde{x}^{(k)} - \tilde{d}^{(k)})}. \quad (2.27)$$

Now using (2.19), we can write

$$\frac{2f'(\tilde{x}^{(k)})}{\tilde{c}^{(k)}(\tilde{x}^{(k)} - \tilde{d}^{(k)})} = -\frac{1}{\tilde{\alpha}^{(k)}}. \quad (2.28)$$

Therefore, after some simplification, equation (2.27) becomes

$$\left(\frac{\tilde{x}^{(k)} - \tilde{d}^{(k)}}{x - \tilde{d}^{(k)}}\right)^2 = 1 - \frac{1}{\tilde{\alpha}^{(k)}}. \quad (2.29)$$

Finally, the required identity (2.23) immediately follows from (2.29) and the above mentioned fact that  $x_k - \tilde{d}^{(k)}$  and  $x - \tilde{d}^{(k)}$  have the same sign in the interval  $\mathcal{I}_k$ . This completes the proof of the theorem.

Let us now define the notion of feasibility for a sequence of asymptotes  $\{\tilde{r}^{(k)}\} := \{\tilde{r}^{(k)}\}_k$ , which we will often use in the sequel.

**Definition 2.3.2** Let  $w$  a function, satisfies (2.10) and (2.11). A sequence of asymptotes  $\{\tilde{r}^{(k)}\}$  is said feasible, if for all  $k \geq 0$ , there exist two real numbers  $L^{(k)}$  and  $U^{(k)}$  satisfying the following:

$$\tilde{r}^{(k)} = \begin{cases} L^{(k)} & \text{if } f'(\tilde{x}^{(k)}) < 0, \text{ and } L^{(k)} < \tilde{x}^{(k)} + \frac{2f'(\tilde{x}^{(k)})}{|f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)})|}, \\ U^{(k)} & \text{if } f'(\tilde{x}^{(k)}) > 0, \text{ and } U^{(k)} > \tilde{x}^{(k)} + \frac{2f'(\tilde{x}^{(k)})}{|f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)})|}. \end{cases} \quad (2.30)$$

It is clear from the above definition that every feasible sequence of asymptotes  $\{\tilde{r}^{(k)}\}$  also satisfies the constraints of type (2.4).

The following proposition, which is easily obtained by a simple algebraic manipulation, shows that the lower bound of the difference between the asymptotes and the current iterate  $\tilde{x}^{(k)}$ , can be estimated as in (2.31).

**Proposition 2.3.3** Let  $\{\tilde{d}^{(k)}\}$  be a sequence of asymptotes, where  $w$  is a continuous function satisfying (2.11) and let the assumptions (2.4) valid. Then,  $\{\tilde{d}^{(k)}\}$  is feasible if and only if

$$\left| \frac{2f'(\tilde{x}^{(k)})}{f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)})} \right| < |\tilde{x}^{(k)} - \tilde{d}^{(k)}|. \quad (2.31)$$

By defining the suitable index set

$$I^{(k)} = \left\{ \begin{array}{l} \left[ \tilde{d}^{(k)}, +\infty \right] \quad \text{if } f'(\tilde{x}^{(k)}) < 0, \\ \left[ -\infty, \tilde{d}^{(k)} \right] \quad \text{if } f'(\tilde{x}^{(k)}) > 0, \end{array} \right\}. \quad (2.32)$$

**Remark 2.3.4** *If we take  $w = 0$ , so the null function is continuous and satisfied (2.11), then the interpolation condition:*

$$(\tilde{f}_w^{(k)})''(\tilde{x}^{(k)}) = |f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)})|.$$

become

$$(\tilde{f}_w^{(k)})''(\tilde{x}^{(k)}) = |f''(\tilde{x}^{(k)})|.$$

and if we assume that  $f''(\tilde{x}^{(k)}) > 0 \quad \forall k \geq 0$  then

$$\tilde{f}_w^{(k)} = \tilde{f}^{(k)}$$

where  $\tilde{f}^{(k)}$  is the approximated function defined by

$$\tilde{f}^{(k)}(x) = a^{(k)} + b^{(k)}(x - \tilde{x}^{(k)}) + \tilde{c}^{(k)} \left( \frac{1}{2} \frac{(\tilde{x}^{(k)} - \tilde{d}^{(k)})^3}{x - \tilde{d}^{(k)}} + \frac{1}{2} (\tilde{x}^{(k)} - \tilde{d}^{(k)})(x - 2\tilde{x}^{(k)} + \tilde{d}^{(k)}) \right) \quad (2.33)$$

$$\begin{aligned} a^{(k)} &= f(\tilde{x}^{(k)}), \\ b^{(k)} &= f'(\tilde{x}^{(k)}), \\ \tilde{c}^{(k)} &= f''(\tilde{x}^{(k)}). \end{aligned} \quad (2.34)$$

and if we take  $\alpha_k = \alpha > 1$ , we get

$$\tilde{d}^{(k)} = \tilde{x}^{(k)} + 2\alpha \frac{f'(\tilde{x}^{(k)})}{f''(\tilde{x}^{(k)})}, \quad \text{with } \alpha > 1 \quad (2.35)$$

We can rewrite  $\tilde{d}^{(k)}$  as

$$\tilde{d}^{(k)} = \begin{cases} L_k < \tilde{x}^{(k)} + 2 \frac{f'(\tilde{x}^{(k)})}{f''(\tilde{x}^{(k)})} & \text{if } f'(\tilde{x}^{(k)}) < 0, \\ U_k > \tilde{x}^{(k)} + 2 \frac{f'(\tilde{x}^{(k)})}{f''(\tilde{x}^{(k)})} & \text{if } f'(\tilde{x}^{(k)}) > 0. \end{cases} \quad (2.36)$$

So this method can be seen as an extension of the method of moving asymptotes with explicit solutions.

We have built a strictly convex approximation function of the objective function  $f$  at  $\tilde{x}^{(k)}$ , which is minimum at  $\tilde{x}_*^{(k)} \in I_k$ . We have now to verify that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  defined by  $\tilde{x}^{(k+1)} = \tilde{x}_*^{(k)}$  converges to the solution  $x_*$  of the initial problem.

### 2.3.1 Convergence study

In this Section, we give the main result of this chapter, that is sufficient conditions on the data (the point  $\tilde{x}^{(0)}$ , the function  $f'$  in a neighborhood of  $\tilde{x}^{(0)}$ , the family  $f''(\tilde{x}^{(k)}), k \geq 0$ ), which guarantee that first derivative of  $f$  vanishes in a neighborhood of  $x^*$ , first, and secondly, the convergence of the method to this zero.

To establish our convergence results, we need the following assumptions. We assume that there exist positive constants  $r, M, C$  and  $\xi < 1$  such that the following assumptions hold:

#### Assumption 1

$$B_r := \left\{ x \in \mathbb{R} : |x - \tilde{x}^{(0)}| \leq r \right\} \subset \Omega.$$

#### Assumption 2

$$\sup_{k \geq 0} |\tilde{x}^{(k)} - \tilde{d}^{(k)}| \leq C.$$

#### Assumption 3

$$\frac{C}{M} \leq \frac{\tilde{c}^{(k)}}{2} |\tilde{x}^{(k)} - \tilde{d}^{(k)}| - |f'(\tilde{x}^{(k)})|.$$

#### Assumption 4

$$\sup_{k \geq 0} \sup_{x \in B} \left| f''(x) - \frac{f'(x^{(k-1)})}{x^{(k-1)} - \tilde{x}^{(k)}} \right| \leq \frac{\xi}{M}.$$

#### Assumption 5

$$0 < |f'(\tilde{x}^{(0)})| \leq \frac{r}{M} (1 - \xi).$$

Assumption 2 enforces the quite natural conditions that are: at each iteration  $k$ , the distance between  $\tilde{x}^{(k)}$  and the interval boundary  $\tilde{d}^{(k)}$  is bounded. Furthermore, Assumption 4 tells us that the coefficient  $\tilde{c}^{(k)}$  does not change too much in a neighborhood of  $\tilde{x}^{(0)}$ , and finally for any  $k$  the functions  $\tilde{c}^{(k)}$  and  $\frac{1}{\tilde{c}^{(k)}}$  have not to change too much in a neighborhood of  $\tilde{x}^{(0)}$ , and  $\tilde{c}^{(k)}$  being sufficiently near  $c^{(0)}$  where  $c^{(0)} = |f''(\tilde{x}^{(0)}) + w(\tilde{x}^{(0)})f'(\tilde{x}^{(0)})|$ . Finally, Assumption 5 only says that  $|f'(\tilde{x}^{(0)})|$  is small enough and that  $f'(\tilde{x}^{(0)})$  is non zero.

Throughout this subsection, we suppose that Assumptions 1-5 hold. The constants  $r, M, C$  and  $\xi < 1$  that appear in the subsequent analysis are always the constants from Assumptions 1-5. Our aim is to show that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  defined by (2.37) converges geometrically to a point  $x^*$  in the sense that  $|\tilde{x}^{(k)} - x^*| \leq \frac{\xi^k}{1 - \xi} |x^{(1)} - x^{(0)}|$ .

**Theorem 2.3.5** *Assume Assumptions 1-5 hold. Let the assumptions of theorem 2.3.1 be valid and let  $\tilde{s}^{(k)}$  be defined by (2.24). Then the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  given by*

$$\tilde{x}^{(k+1)} = \tilde{d}^{(k)} + (\tilde{x}^{(k)} - \tilde{d}^{(k)})\sqrt{\tilde{s}^{(k)}} \quad (2.37)$$

*is completely contained in the interval  $B_r$ , and converges to the unique zero of  $f'$  in  $B_r$ .*

Before we embark on the proof of Theorem 2.3.5, we first prove some technical lemmas.

**Lemma 2.3.6** *Let Assumption 2 and 3 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be as defined in Theorem 2.3.5. Then, for any positive integer  $k$  the following inequality holds.*

$$\left| \tilde{x}^{(k)} - \tilde{x}^{(k-1)} \right| \leq M \left| f'(\tilde{x}^{(k-1)}) \right|. \quad (2.38)$$

**Proof 2** *Let us fix a positive integer  $k$ . Using (2.37) we may write*

$$\begin{aligned} \tilde{x}^{(k)} - \tilde{x}^{(k-1)} &= \tilde{d}^{(k-1)} + (\tilde{x}^{(k-1)} - \tilde{d}^{(k-1)})\sqrt{\tilde{s}^{(k-1)}} - \tilde{x}^{(k-1)} \\ &= (\tilde{x}^{(k-1)} - \tilde{d}^{(k-1)}) \left( \sqrt{\tilde{s}^{(k-1)}} - 1 \right) \end{aligned} \quad (2.39)$$

Now, from (2.20) we have

$$\tilde{s}^{(k-1)} > 1, \quad (k > 1), \quad (2.40)$$

we then immediately deduce

$$\sqrt{\tilde{s}^{(k-1)}} < \tilde{s}^{(k-1)}.$$

Therefore, by (2.39), we arrive at

$$\left| \tilde{x}^{(k)} - \tilde{x}^{(k-1)} \right| \leq \left| \tilde{x}^{(k-1)} - \tilde{d}^{(k-1)} \right| \left( \tilde{s}^{(k-1)} - 1 \right), \quad (2.41)$$

this, after some manipulations, leads

$$\left| \tilde{x}^{(k)} - \tilde{x}^{(k-1)} \right| \leq \left| \tilde{x}^{(k-1)} - \tilde{d}^{(k-1)} \right| \frac{|f'(x^{(k-1)})|}{\frac{\tilde{c}^{(k-1)}}{2} \left| \tilde{x}^{(k-1)} - \tilde{d}^{(k-1)} \right| - |f'(\tilde{x}^{(k-1)})|}.$$

Finally, by Assumptions 2 and 3 we get the required inequality in (2.38).

In order to prove that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  converges geometrically, we need some further preparatory results.

**Lemma 2.3.7** *Let Assumption 4 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be defined as in Theorem 2.3.5. Then, for any positive  $k$  the following inequality holds.*

$$\left| f'(\tilde{x}^{(k)}) \right| \leq \frac{\xi}{M} \left| \tilde{x}^{(k)} - \tilde{x}^{(k-1)} \right|. \quad (2.42)$$

**Proof 3** Fix a positive integer  $k$ . Let us define  $\tilde{t}^{(k-1)}$  by

$$\tilde{t}^{(k-1)} := \frac{\tilde{c}^{(k-1)}}{2} (\tilde{x}^{(k-1)} - \tilde{d}^{(k-1)}) - f'(\tilde{x}^{(k-1)}), \quad (2.43)$$

and the auxiliary function  $\varphi : B \rightarrow \mathbb{R}$  as follows:

$$\varphi(x) := f'(x) - \frac{f'(\tilde{x}^{(k-1)})}{\frac{1}{2}c^{(k-1)}(\tilde{x}^{(k)} - \tilde{x}^{(k-1)})} h(x),$$

where

$$h(x) := -f'(\tilde{x}^{(k-1)}) - \frac{\tilde{c}^{(k-1)}}{2} (x - \tilde{x}^{(k)} + \tilde{x}^{(k-1)} - \tilde{d}^{(k-1)}) - \tilde{t}^{(k-1)}.$$

Using (2.43), it is easily checked that  $\varphi$  satisfies

$$\begin{aligned} \varphi(\tilde{x}^{(k-1)}) &= 0, \\ \varphi(\tilde{x}^{(k)}) &= f'(\tilde{x}^{(k)}). \end{aligned}$$

Then, from the mean-value theorem and Assumption 4 we get

$$\begin{aligned} |f'(\tilde{x}^{(k)})| &\leq \sup_{x \in B_r} \left| f''(x) - \frac{f'(\tilde{x}^{(k-1)})}{\tilde{x}^{(k-1)} - \tilde{x}^{(k)}} \right| |\tilde{x}^{(k)} - \tilde{x}^{(k-1)}| \\ &\leq \frac{\xi}{M} |\tilde{x}^{(k)} - \tilde{x}^{(k-1)}|. \end{aligned} \quad (2.44)$$

This shows that the required inequality (2.42) holds true for any positive integer  $k$ .

The next result shows that for all  $k$  the iterate  $\tilde{x}^{(k)}$  remains in the interval  $B_r$ .

**Lemma 2.3.8** Let Assumption 2-4 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be as defined in Theorem 2.3.5. Assume that the starting point  $\tilde{x}_0$  belongs to the interval  $B_r$ , where  $r$  is defined in Assumption 1. Then, all terms of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  lie inside the interval  $B_r$ .

**Proof 4** Indeed, combining inequalities (2.38) and (2.42) of Lemmas 2.3.6 and 2.3.7 respectively, we immediately obtain

$$|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}| \leq \xi |\tilde{x}^{(k-1)} - \tilde{x}^{(k-2)}| \leq \dots \leq \xi^{k-1} |\tilde{x}^{(1)} - \tilde{x}^{(0)}|, \quad (2.45)$$

and therefore we have

$$\begin{aligned} |\tilde{x}^{(k)} - \tilde{x}^{(0)}| &\leq \sum_{l=1}^k |\tilde{x}^{(l)} - \tilde{x}^{(l-1)}| \\ &\leq \left( \sum_{l=1}^k \xi^{l-1} \right) |\tilde{x}^{(1)} - \tilde{x}^{(0)}| \\ &\leq \frac{|\tilde{x}^{(1)} - \tilde{x}^{(0)}|}{1 - \xi}. \end{aligned} \quad (2.46)$$

Finally, applying inequality (2.38) for  $k = 1$  and using Assumption 5 we immediately get

$$\begin{aligned} |\tilde{x}^{(k)} - \tilde{x}^{(0)}| &\leq \frac{M}{1 - \xi} |f'(\tilde{x}^{(0)})| \\ &\leq r, \end{aligned} \quad (2.47)$$

which shows that each  $\tilde{x}^{(k)}$  belongs to  $B_r$ .

As a consequence of the previous three lemmas, we are now in a position to prove Theorem 2.3.5.

*Proof of Theorem 2.3.5* Since the entire sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  remains in the (closed) interval  $B_r$  by Lemma 2.3.8, every limit point of this sequence belongs to this set, too. Hence, it remains to show that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  converges. To this end, we first note that, for  $k \geq 0$  and  $l \geq 0$ , we have

$$\begin{aligned} |\tilde{x}^{(k+l)} - \tilde{x}^{(k)}| &\leq \sum_{v=0}^{l-1} |\tilde{x}^{(k+v+1)} - \tilde{x}^{(k+v)}| \\ &\leq \xi^k \sum_{v=0}^{l-1} \xi^v |\tilde{x}^{(1)} - \tilde{x}^{(0)}| \\ &\leq \frac{\xi^k}{1 - \xi} |\tilde{x}^{(1)} - \tilde{x}^{(0)}|, \end{aligned} \quad (2.48)$$

then the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  is a Cauchy sequence. Being Cauchy in  $B_r$  (closed interval in  $\mathbb{R}$ ), it has a limit,  $x^*$ , in  $B_r$ . Now, thanks to the continuity of  $f'$  on  $B_r$ , (2.42) the continuity of  $f'$  on  $B$  and the convergence the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  imply

$$|f'(\tilde{x}^*)| = \lim_{k \rightarrow +\infty} |f'(\tilde{x}^{(k)})| \leq \frac{\xi}{M} \lim_{k \rightarrow +\infty} |\tilde{x}^{(k)} - \tilde{x}^{(k-1)}| = 0, \quad (2.49)$$

and then  $f'(\tilde{x}^*) = 0$ .

Passing to the limit for  $l$  tending to  $\infty$  in (2.48), we deduce that

$$|\tilde{x}^{(k)} - \tilde{x}^*| \leq \frac{\xi^k}{1 - \xi} |\tilde{x}^{(1)} - \tilde{x}^{(0)}|, \quad (2.50)$$

which shows the geometric convergence of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  to  $\tilde{x}^*$ .

We are now in a position to prove that  $f'$  has a unique zero in  $B_r$ . To this end, we proceed by contradiction, assuming that  $f'$  has another zero  $\tilde{y}^* \in B$ . Let us introduce the auxiliary function

$$\lambda(x) = \frac{\tilde{x}^{(1)} - \tilde{x}^{(0)}}{f'(\tilde{x}^{(0)})} \left( f'(x) - \frac{f'(\tilde{x}^{(0)})}{\tilde{x}^{(0)} - \tilde{x}^{(1)}} (x - \tilde{x}^*) \right),$$

which satisfies  $\lambda(\tilde{x}^*) = 0$  and  $\lambda(\tilde{y}^*) = \tilde{y}^* - x^*$ . Therefore, applying (2.38) for  $k = 1$ , it follows from the mean value theorem and Lemma 2.3.6, inequality (2.38) for  $k = 1$ ,

$$\begin{aligned} |\tilde{x}^* - \tilde{y}^*| &\leq \left| \frac{\tilde{x}^{(1)} - \tilde{x}^{(0)}}{f'(\tilde{x}^0)} \right| \sup_{x \in B} \left| f''(x) - \frac{f'(\tilde{x}_0)}{\tilde{x}^0 - \tilde{x}^{(1)}} \right| |\tilde{x}^* - \tilde{y}^*| \\ &\leq M \frac{\xi}{M} |\tilde{x}^* - \tilde{y}^*| \\ &\leq \xi |\tilde{x}^* - \tilde{y}^*|. \end{aligned} \tag{2.51}$$

This yields  $x^* = \tilde{y}^*$  since  $\xi < 1$ . Thus, the theorem is proved.

### 2.3.2 Description of algorithm

The results of the previous section may be used to construct the following algorithm.

---

**Algorithm 1** Modified Method of Moving Asymptotes

---

- 1: Data:  $\tilde{x}^{(0)}$ ,  $w$ ,  $\tilde{\alpha}^n > 1$ ,  $\varepsilon$
  - 2:  $k = 0$
  - 3: REPEAT
  - 4:  $\tilde{c}^{(k)} = \left| f''(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f'(\tilde{x}^{(k)}) \right|$ ,
  - 5:  $\tilde{d}^{(k)} = \tilde{x}^{(k)} + 2\tilde{\alpha}^{(k)} \frac{f'(\tilde{x}^{(k)})}{\tilde{c}^{(k)}}$ ,
  - 6:  $\tilde{s}^{(k)} = \frac{|\tilde{x}^{(k)} - \tilde{d}^{(k)}|^3}{|\tilde{x}^{(k)} - \tilde{d}^{(k)}| - \frac{2|f'(\tilde{x}^{(k)})|}{\tilde{c}^{(k)}}}$ ,
  - 7:  $\tilde{x}^{(k+1)} = \tilde{d}^{(k)} + \text{sign}(\tilde{x}^{(k)} - \tilde{d}^{(k)})\sqrt{\tilde{s}^{(k)}}$ ,
  - 8: while  $\left| f'(\tilde{x}^{(k)}) \right| > \varepsilon$ .
- 

## 2.4 Numerical examples

We employ the present method (designated as present) to solve some nonlinear, non-convex optimization problems and compare it with the [3] method, Newton's method and the BFGS method using two kinds of weight functions and four test functions:

$$\begin{aligned} f_1(x) &= \frac{1}{3}(\sin^3 x - x^3) + x, \\ f_2(x) &= \frac{1}{2}\exp(x^2) + \frac{1}{2}\left(x - \frac{1}{2}\sin(2x)\right) + 3\sin x + 5x, \\ f_3(x) &= -\left(\frac{1}{3}x^3 + \frac{5}{2}x^2 + 3x - \exp(x)\right), \\ f_4(x) &= \frac{(x-1)^4}{4} - 2x + 1. \end{aligned}$$

Numerical results are summarized in Tables 2.1 and 2.2, where for each weight function  $w$ , we present the objective functions, the starting points, the methods used, the number of iterations  $N$  to obtain the objective value of the obtained optimal solution  $x^{(N)}$  and  $f(x)$  at  $x^{(N)}$ . We use the following stopping criteria for computer programs:  $|f'(\tilde{x}^{(N)})| < \varepsilon$ , (the absolute value of the derivative of the function is less than or equal to the tolerance). For numerical illustrations we used different values of  $\varepsilon$ . Therefore, when the stopping criterion is satisfied,  $\tilde{x}_* = \tilde{x}^{(N)}$  is taken as the optimal solution. In Tables 2.1 and 2.2 div. means that the stopping criteria is not satisfied. Let  $k$  be the iteration index, then we choose the moving asymptotes as follows:

The test results in Tables 2.1 and 2.2 show that for all of the functions we tested, the present method is better than the [3] method, Newton's method and the BFGS Method. It always converges even if the starting point is very far from the true solution, and it never requires more iterations than the other three methods. In addition, as we can see, it may converge when these latter div.erge simultaneously. These characteristics give a strong advantage over these other methods.

Function	Method	$N$	$x_N$	$f(x_N) \simeq \min f(x)$	$\varepsilon$
$f_1(x)$ $\tilde{x}^{(0)} = 10^{-12}$	.	.	.	.	$10^{-14}$
	Newton	div.	.	.	.
	BFGS	div.	.	.	.
	[3]	div.	.	.	.
	Present	6	-1.156436	$-896.585243e - 003$	.
$\tilde{x}^{(0)} = -\frac{1}{4}$	.	.	.	.	.
	Newton	13	.	.	.
	BFGS	13	.	.	.
	[3]	5	-1.156436	$-896.585243e - 003$	.
	Present	5	.	.	.
$f_2(x)$ $\tilde{x}^{(0)} = 0.25$	.	.	.	.	$10^{-15}$
	Newton	div.	.	.	.
	BFGS	div.	.	.	.
	[3]	div.	.	.	.
	Present	8	-1.231394	-7.166822	.
$\tilde{x}^{(0)} = -10$	.	.	.	.	.
	Newton	110	.	.	.
	BFGS	110	.	.	.
	[3]	114	-1.231394	-7.166822	.
	Present	107	.	.	.
$f_3(x)$ $\tilde{x}^{(0)} = -2.5$	.	.	.	.	$10^{-16}$
	.	.	.	.	.



.	Newton	div.	.	.	.
.	BFGS	div.	.	.	.
.	[3]	div.	.	.	.
.	Present	20	-4.306510	-6.809174	.
$\tilde{x}^{(0)} = 12$	.	.	.	.	.
.	Newton	17	.	.	.
.	BFGS	17	.	.	.
.	[3]	31	3.482467 *—* $\zeta$	-2.230440e+001	.
.	Present	12	.	.	.

Table 2.1 Numerical comparisons of the [3] method, Newton’s method, the BFGS Method and the present method. Here  $w(x) = (1 + |x|)^{1/2}exp(-2|x|)$ .

Function	$f(x_0)$	Method	$N$	$x_N$	$f(x_N) \simeq min f(x)$
$f_1(x)$	.	.	.	.	.
$\tilde{x}^{(0)} = -62 * 10^{100}$	79.44e+303	.	.	.	.
.	.	Newton	div.	.	.
.	.	BFGS	div.	.	.
.	.	[3]	div.	.	.
.	.	Present	241	-1.156436	-896.585243e - 003
$\tilde{x}^{(0)} = 40 * 10^{60}$	-21.33e+183	.	.	.	.
.	.	Newton	div.	.	.
.	.	BFGS	div.	.	.
.	.	[3]	div.	.	.
.	.	Present	152	-1.156436	-896.585243e - 003
$\tilde{x}^{(0)} = -30 * 10^{10}$	9.00e+033	.	.	.	.
.	.	Newton	44	.	.
.	.	BFGS	44	.	.
.	.	[3]	div.	-1.156436	-896.585243e - 003
.	.	Present	40	.	.
$f_2(x)$	.	.	.	.	.
$\tilde{x}^{(0)} = 26$	19.14e+292	.	.	.	.
.	.	Newton	div.	.	.
.	.	BFGS	div.	.	.
.	.	[3]	div.	.	.
.	.	Present	556	-1.231394	-7.166822
$\tilde{x}^{(0)} = 10$	13.44e+042	.	.	.	.
.	.	Newton	div.	.	.
.	.	BFGS	div.	.	.

.	.	[3]	div.	-1.231394	-7.166822
.	.	Present	136	.	.
<hr/>					
$f_3(x)$	.	.	.	.	.
$\tilde{x}^{(0)} = -30 * 10^{100}$	9.00e+303	.	.	.	.
.	.	Newton	340	.	.
.	.	BFGS	340	.	.
.	.	[3]	div.	.	.
.	.	Present	238	-1.156436	-896.585243e - 003
$\tilde{x}^{(0)} = -21 * 10^{50}$	30.87e+152	.	.	.	.
.	.	Newton	173	.	.
.	.	BFGS	173	.	.
.	.	[3]	div.	.	.
.	.	Present	127	-1.156436	-896.585243e - 003
$\tilde{x}^{(0)} = -30 * 10^{10}$	8.99e+033	.	.	.	.
.	.	Newton	44	.	.
.	.	BFGS	44	.	.
.	.	[3]	div.	-1.156436	-896.585243e - 003
.	.	Present	40	.	.
<hr/>					
$f_4(x)$	.	.	.	.	.
$\tilde{x}^{(0)} = 20 * 10^{70}$	4.00e+284	.	.	.	.
.	.	Newton	div.	.	.
.	.	BFGS	div.	.	.
.	.	[3]	div.	.	.
.	.	Present	428	2.259921	-2.889881
$\tilde{x}^{(0)} = 40 * 10^{40}$	6.40e+165	.	.	.	.
.	.	Newton	240	.	.
.	.	BFGS	240	.	.
.	.	[3]	div.	.	.
.	.	Present	184	2.259921	-2.889881

Table 2.2 Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present paper. Here  $w(x) = (1 + |x|)^{-4} \exp(-10 |x|^{0.5}) * \log(e + |x|)^{10}$ .



## Chapter 3

# A GLOBALLY CONVERGENT MULTIVARIATE VERSION OF THE METHOD OF MOVING ASYMPTOTES

### 3.1 Motivation and theoretical justification

We consider some new iterative methods for solving the unconstrained optimization problem: Find  $x_* \in \Omega$  such that

$$f(x_*) = \min_{x \in \Omega} f(x), \quad (3.1)$$

where  $\Omega$  is an open subset of  $\mathbb{R}^n$  and  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  is a given non-linear real-valued objective function, typically twice continuously differentiable, which could be non-convex. In order to evaluate the merit of using second order information an extension of the method of moving asymptotes, that accounts for the curvatures, was proposed in [3]. Let us put these things a bit more precisely. Throughout, we assume that  $\nabla f$  does not vanish at a given suitable initial point  $x^{(0)} \in \Omega$ , that is  $\nabla f(x^{(0)}) \neq 0_{\mathbb{R}^n}$ , since if this is not the case we have nothing to solve. Starting from the initial design point  $x^{(0)}$  the iterates  $x^{(k)}$  are computed successively by solving sub-problems of the form: Find  $x^{(k+1)}$  such that

$$f^{(k)}(x^{(k+1)}) = \min_{x \in \Omega} f^{(k)}(x), \quad (3.2)$$

where the approximating function  $f^{(k)}$  of the objective function  $f$  at the  $k$ -th iteration has the following form:

$$f^{(k)}(x) = \sum_{j=1}^n \left( \frac{\alpha_-^{(k)}}{x_j - L_j^{(k)}} + \frac{\alpha_+^{(k)}}{U_j^{(k)} - x_j} \right) + \langle \beta_-^{(k)}, x - L^{(k)} \rangle + \langle \beta_+^{(k)}, U^{(k)} - x \rangle + \gamma^{(k)}, \quad (3.3)$$

The coefficients  $\beta_-^{(k)}$ ,  $\beta_+^{(k)}$ ,  $L^{(k)}$  and  $U^{(k)}$  are some chosen parameters given by

$$\begin{aligned} \beta_-^{(k)} &= \left( (\beta_-^{(k)})_1, (\beta_-^{(k)})_2, \dots, (\beta_-^{(k)})_n \right), \\ \beta_+^{(k)} &= \left( (\beta_+^{(k)})_1, (\beta_+^{(k)})_2, \dots, (\beta_+^{(k)})_n \right), \\ L^{(k)} &= \left( L_1^{(k)}, L_2^{(k)}, \dots, L_n^{(k)} \right), \\ U^{(k)} &= \left( U_1^{(k)}, U_2^{(k)}, \dots, U_n^{(k)} \right), \end{aligned}$$

and  $\gamma^{(k)} \in \mathbb{R}$ . They represent the unknown parameters that need to be computed based on the available information. In contrast with the classical Newton method, here the approximation functions  $f^{(k)}$  are of the form of a *linear* function plus a *rational* function. For each iteration, the approximate parameters  $\alpha_-^{(k)}$ ,  $\alpha_+^{(k)}$ ,  $\beta_-^{(k)}$ ,  $\beta_+^{(k)}$  and  $\gamma^{(k)}$  used in equation (3.3) are determined in such a way that the following set of interpolation conditions are satisfied:

$$f^{(k)}(x^{(k)}) = f(x^{(k)}), \quad (3.4)$$

$$f_{,j}^{(k)}(x^{(k)}) = f_{,j}(x^{(k)}), \quad j = 1, \dots, n. \quad (3.5)$$

$$f_{,,jj}^{(k)}(x^{(k)}) = f_{,,jj}(x^{(k)}), \quad (3.6)$$

In order to ensure that the functions  $\tilde{f}^{(k)}$  have suitable proprieties discussed in [3], the author have assumed that the following conditions are satisfied for all  $k$ :

$$\begin{aligned} \left( \alpha_-^{(k)} \right)_j = \left( \beta_-^{(k)} \right)_j = 0 & \text{ if } f_{,j}(x^{(k)}) > 0, \\ \left( \alpha_+^{(k)} \right)_j = \left( \beta_+^{(k)} \right)_j = 0 & \text{ if } f_{,j}(x^{(k)}) < 0, \end{aligned} \quad j = 1, \dots, n. \quad (3.7)$$

It follows from the above identities that  $\alpha_-^{(k)}$ ,  $\alpha_+^{(k)}$ ,  $\beta_-^{(k)}$ ,  $\beta_+^{(k)}$  and  $\gamma^{(k)}$  are given by

$$\left( \alpha_-^{(k)} \right)_j = \begin{cases} \frac{1}{2} S_{jj}^{(k)} (x_j^{(k)} - L_j^{(k)})^3 & \text{if } f_{,j}(x^{(k)}) < 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.8)$$

$$\left(\alpha_+^{(k)}\right)_j = \begin{cases} \frac{1}{2}S_{jj}^{(k)} \left(U_j^{(k)} - x_j^{(k)}\right)^3 & \text{if } f_{:,j}(x^{(k)}) > 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.9)$$

$$\left(\beta_-^{(k)}\right)_j = \begin{cases} f_{:,j}(x^{(k)}) + \frac{\left(\alpha_-^{(k)}\right)_j}{\left(x_j^{(k)} - L_j^{(k)}\right)^2} & \text{if } f_{:,j}(x^{(k)}) < 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.10)$$

$$\left(\beta_+^{(k)}\right)_j = \begin{cases} f_{:,j}(x^{(k)}) - \frac{\left(\alpha_+^{(k)}\right)_j}{\left(U_j^{(k)} - x_j^{(k)}\right)^2} & \text{if } f_{:,j}(x^{(k)}) > 0 \\ 0 & \text{otherwise,} \end{cases} \quad (3.11)$$

and hence  $\gamma^{(k)}$  is given by

$$\gamma^{(k)} = f(x^{(k)}) - \sum_{j=1}^n \left( \frac{\left(\alpha_-^{(k)}\right)_j}{x_j^{(k)} - L_j^{(k)}} + \frac{\left(\alpha_+^{(k)}\right)_j}{U_j^{(k)} - x_j^{(k)}} \right) - \langle \beta_-^{(k)}, x^{(k)} - L^{(k)} \rangle - \langle \beta_+^{(k)}, U^{(k)} - x^{(k)} \rangle. \quad (3.12)$$

In (3.8) and (3.9), the coefficients  $S_{jj}^{(k)}$  were simply chosen such that

$$S_{jj}^{(k)} := \frac{e^{(k)}}{\|x^{(k)} - x^{(k-1)}\|^2} \approx f_{:,jj}(x^{(k)}), \quad (3.13)$$

where

$$e^{(k)} := \langle \nabla f(x^{(k)}) - \nabla f(x^{(k-1)}), x^{(k)} - x^{(k-1)} \rangle. \quad (3.14)$$

The authors have also assumed in [3] that the objective function should fulfill for each iteration  $k$  the following conditions:

$$f_{:,jj}(x^{(k)}) > 0, \quad j = 1, \dots, n. \quad (3.15)$$

This is a real weakness of this approach, which drastically limits its application. Hence, this method is very restrictive and also has the following disadvantages :

- It needs good initial solution  $x_*$  close to the exact solution.
- It converges slowly, in many cases, to the optimum  $x_*$ .
- It does not always converge.
- Its performance degrades when it applied to non convex functions.
- Incapable of handling non-separability.

The purpose of this chapter is three-fold. First, we develop an extension of our previous paper [37] in a multivariate setting. Second, we propose a modified version of the [3] method by removing the restrictive condition (3.15) on the objective functions. We will concentrate on the method of moving asymptotes since it is considered to belong to the most efficient methods. We believe that similar ideas can be developed for the other members of of several methods for solving minimization problems. Third, we show how the proposed

algorithm can be modified in order that the technique can be applied to a fairly large class of objective functions. Moreover, it is shown, as will be proved below, that the new method converges geometrically. Comparative numerical simulations are conducted to show the success of the proposed extension for various kinds of different test functions. The results suggest that this latter is significantly faster compared to the [3] method, Newton's method or the BFGS method on all test functions and it can succeed where these latter diverge simultaneously. It also has the advantage that, under appropriate conditions, global convergence of the algorithm is guaranteed.

The remainder of this chapter is organized as follows. In Section 3.2, we develop a modified moving asymptotes method. Section 3.3 contains all technical results that are essential to show that our method is guaranteed to converge geometrically. In Section 3.4, we will concentrate on how our algorithm can be extended to the general setting where the objective function is nonconvex and non-separable. The point is not to give a definite answer to this problem but, rather, to clarify and understand how our algorithm can be used for this general situation. The test problems are Wood's, Powell's and Branin's functions. They are all documented in [45]. In section 3.5, various numerical experiments conduct to confirm our theoretical finding. In addition, the comparison with the models considered is also made and we conclude in Section 3.6.

## 3.2 A special modified multivariate version of moving asymptotes method

Throughout this section we assume that  $w$  is a function satisfying the following conditions:

$$w \text{ is a real-valued function, defined and continuous on } \mathbb{R}^n, \quad (3.16)$$

$$\lim_{|x| \rightarrow +\infty} w(x) = 0. \quad (3.17)$$

The general modification of moving asymptotes method that we examine herein may be described as follows: Given the iteration point  $\tilde{x}^{(k)}$  (at iteration  $k$ ).

- Our approach is to iteratively approximate at the  $k$ -th iteration the objective function by the approximating function  $\tilde{f}_w^{(k)}$  where:

$$f(x) \approx \tilde{f}_w^{(k)}(x) = \sum_{j=1}^n \frac{\tilde{c}_j^{(k)}}{x_j - \tilde{d}_j^{(k)}} + \langle \tilde{b}^{(k)}, x - \tilde{d}^{(k)} \rangle + \tilde{a}^{(k)} \quad (3.18)$$

and the coefficients  $\tilde{b}^{(k)}, \tilde{c}^{(k)}, \tilde{d}^{(k)}$  are some chosen parameters given by

$$\begin{aligned} \tilde{b}^{(k)} &= (\tilde{b}_1^{(k)}, \dots, \tilde{b}_n^{(k)}), \\ \tilde{c}^{(k)} &= (\tilde{c}_1^{(k)}, \dots, \tilde{c}_n^{(k)}), \\ \tilde{d}^{(k)} &= (\tilde{d}_1^{(k)}, \dots, \tilde{d}_n^{(k)}), \end{aligned}$$

and  $\tilde{a}^{(k)} \in \mathbb{R}$ . A straightforward calculation shows that the first and the second-order partial derivatives of  $\tilde{f}_w^{(k)}$  have the following expressions:

$$(\tilde{f}_w^{(k)})_{,j}(x) = \tilde{b}_j^{(k)} - \frac{\tilde{c}_j^{(k)}}{(x_j - \tilde{d}_j^{(k)})^2}, \quad j = 1, \dots, n, \quad (3.19)$$

$$(\tilde{f}_w^{(k)})_{,,jj}(x) = \frac{2\tilde{c}_j^{(k)}}{(x_j - \tilde{d}_j^{(k)})^3}, \quad j = 1, \dots, n, \quad (3.20)$$

and since the function  $\tilde{f}_w^{(k)}$  is separable, therefore if  $i \neq j$ , we have:

$$(\tilde{f}_w^{(k)})_{,,ij} = 0, \quad i, j = 1, \dots, n. \quad (3.21)$$

- The approximating function  $\tilde{f}_w^{(k)}$  is first order approximations of the original function  $f$  at the current iteration point  $\tilde{x}^{(k)}$ , i.e.,

$$\tilde{f}_w^{(k)}(\tilde{x}^{(k)}) = f(\tilde{x}^{(k)}), \quad (3.22)$$

$$(\tilde{f}_w^{(k)})_{,j}(\tilde{x}^{(k)}) = f_{,j}(\tilde{x}^{(k)}), \quad j = 1, \dots, n. \quad (3.23)$$

In addition to the above conditions (3.22) and (3.23), the approximating function should satisfy the following more general condition (3.24) instead of (3.6):

$$(\tilde{f}_w^{(k)})_{,,jj}(\tilde{x}^{(k)}) = \left| f_{,,jj}(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f_{,j}(\tilde{x}^{(k)}) \right|. \quad (3.24)$$

Consequently, in the present situation, the approximate parameters  $\tilde{a}^{(k)}$ ,  $\tilde{b}^{(k)}$  and  $\tilde{c}^{(k)}$  can be expressed in the following forms:

$$\tilde{a}^{(k)} = f(\tilde{x}^{(k)}) - \sum_{j=1}^n \frac{\tilde{x}_j^{(k)}}{\tilde{c}_j^{(k)} - \tilde{d}_j^{(k)}} - \langle \tilde{b}^{(k)}, \tilde{x}^{(k)} - \tilde{d}^{(k)} \rangle, \quad (3.25)$$

$$\tilde{b}_j^{(k)} = f_{,j}(\tilde{x}^{(k)}) + \frac{\tilde{c}_j^{(k)}}{(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})^2}, \quad (3.26)$$

$$\tilde{c}_j^{(k)} = \frac{\left| f_{,,jj}(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f_{,j}(\tilde{x}^{(k)}) \right|}{2} \left( \tilde{x}_j^{(k)} - \tilde{d}_j^{(k)} \right)^3. \quad (3.27)$$

Furthermore, in order to fully determine an explicit expression for the approximating function  $\tilde{f}_w^{(k)}$ , the parameter  $\tilde{d}^{(k)}$  is chosen such that

$$\tilde{d}_j^{(k)} = \tilde{x}_j^{(k)} + 2\tilde{\alpha}_j^{(k)} \frac{f_{,j}(\tilde{x}^{(k)})}{\left| f_{,,jj}(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f_{,j}(\tilde{x}^{(k)}) \right|}, \quad (3.28)$$

for simplicity, we put

$$\gamma_j^{(k)} = \left| f_{,,jj}(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f_{,j}(\tilde{x}^{(k)}) \right| > 0. \quad (3.29)$$

So we can rewrite

$$\tilde{d}_j^{(k)} = \tilde{x}_j^{(k)} + 2\tilde{\alpha}_j^{(k)} \frac{f_{,j}(\tilde{x}^{(k)})}{\gamma_j^{(k)}}, \quad (3.30)$$

where  $\{\tilde{\alpha}^{(k)}\}_k := \{(\tilde{\alpha}_1^{(k)}, \dots, \tilde{\alpha}_n^{(k)})\}_k$  is a sequence of  $\mathbb{R}^n$  with

$$\tilde{\alpha}_j^{(k)} > 1, k \in \mathbb{N} \quad \text{and} \quad j = 1, \dots, n. \quad (3.31)$$



Different rules for how to choose these values (and possible weight functions in (3.24)) will be provided later. We note that our method does not use the interpolation condition (3.6), but instead we have incorporated a first- and second-order information, as given in (3.27). Moreover, in particular, if you take  $w = 0$  and at each iteration condition (3.15) is fulfilled, then our iterative scheme obviously reduces to the one introduced in [3]. Hence, subsequent iterations of the [3] method are essentially the same as the approximating function (3.18), except that in the proposed approximating function  $\tilde{f}_w^{(k)}$  the parameters  $c^{(k)}$  and  $d^{(k)}$  are replaced by those given in (3.27) and (3.30) respectively. Thus our scheme starts with some guess point  $\tilde{x}^{(0)}$  and generates successive iterates by

$$f(\tilde{x}^{(k+1)}) \leftarrow \tilde{f}_w^{(k)}(\tilde{x}^{(k+1)}) = \min_{x \in \Omega} \tilde{f}_w^{(k)}(x). \quad (3.32)$$

For the sake of notation simplicity, we have removed the index  $w$  in the iterative sequence  $\tilde{x}_w^{(k)}$ .

One of the key ingredient of the proposed approach is to work with (3.27) instead of (3.6). This modification will play an important role in the analysis of the proposed modified algorithm. Indeed, there are several good reasons for this choice. First, as mentioned above, the main reason is that this allows us to apply our method to a large class of objective functions. Second, there are also important advantages from the numerical point of view: many experimental results will confirm that the iterative scheme based on our modification version (3.27) can yield significantly fewer iterations than the [3] method, Newton's method or the BFGS method. Furthermore, in contrast to these three approaches, our method converges even if the starting point is very far from the true solution. In addition, as we will see, the key features of the present method are:

- It does not require us to build a good initial solution close to the exact solution.
- It converges geometrically for a large class of functions  $w$  that satisfy conditions (3.16) and (3.17).
- It will succeed where the [3] method, Newton's method and the BFGS Method break down.

Newton's method and the BFGS Method have a well-studied convergence theory that guarantees the convergence to a solution under a standard set of assumptions. For these and other their variants, the interested reader should consult one of the many excellent books on this subject [15, pp. 48–75] and [34, pp. 75–89]. We refer the readers to [3] and the references therein for the method of moving asymptotes.

### 3.3 Convergence Analysis

We start this section with a result concerning an explicit expression for the iterative sequence  $\{\tilde{x}^{(k)}\}_k$  generated by the approximating function  $\tilde{f}_w^{(k)}$ , as given in (3.18). Here, we continue to denote by  $\tilde{c}^{(k)}$ ,  $\tilde{d}^{(k)}$  and  $\tilde{\alpha}^{(k)}$  the coefficients given by (3.27), (3.30) and (3.31) respectively. Note that condition (3.31), imposed on the parameters  $\tilde{\alpha}^{(k)}$ , is crucial since it will guarantee strict convexity of the approximating function  $\tilde{f}_w^{(k)}$ . For brevity, in the following we use the notation:

$$\mathcal{S}^{(k)} = \mathcal{S}_1^{(k)} \times \dots \times \mathcal{S}_n^{(k)}, \quad (3.33)$$

with

$$\mathcal{S}_i^{(k)} = \left] -\infty, \tilde{d}_i^{(k)} \right[ \cup \left] \tilde{d}_i^{(k)}, +\infty \right[, \quad i = 1, \dots, n. \quad (3.34)$$

Now we are in position to state the first main result.

**Theorem 3.3.1** *With the above notation, let  $\Omega \subset \mathbb{R}^n$  be an open subset, a given twice continuously differentiable function  $f$  in  $\Omega$ ,  $\tilde{x}^{(0)} \in \Omega$  and  $\tilde{x}^{(k)}$  being respectively the initial and a current point of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$ . Then, for each  $k > 0$  the approximating function  $\tilde{f}_w^{(k)}$  defined by (3.18) is a strictly convex function on  $\mathcal{I}^{(k)}$ . In addition, it has an unique minimum at*

$$\tilde{x}^{(k+1)} \leftarrow \tilde{x}_*^{(k)} = \tilde{d}^{(k)} + G^{(k)} \quad (3.35)$$

where  $G^{(k)} = (G_1^{(k)}, \dots, G_n^{(k)})$  with

$$G_j^{(k)} = (\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}) \sqrt{\tilde{s}_j^{(k)}} \quad (3.36)$$

$$\tilde{s}_j^{(k)} = \frac{\tilde{\alpha}_j^{(k)}}{\tilde{\alpha}_j^{(k)} - 1}. \quad (3.37)$$

**Proof 5** *The main ingredient here is a suitable application of the condition (3.31) imposed on the coefficient  $\tilde{d}^{(k)}$ . We first start by showing that the approximating function  $\tilde{f}_w^{(k)}$  is well defined and strictly convex in  $\mathcal{I}^{(k)}$ . To this end we prove that  $\nabla^2 \tilde{f}_w^{(k)}$  is positive semidefinite in  $\mathcal{I}^{(k)}$ . Indeed, a simple calculation reveals that*

$$\nabla^2 \tilde{f}_w^{(k)}(x) = \begin{pmatrix} \gamma_1^{(k)} \left( \frac{\tilde{x}_1^{(k)} - \tilde{d}_1^{(k)}}{x_1 - \tilde{d}_1^{(k)}} \right)^3 & 0 & \dots & 0 \\ 0 & \gamma_2^{(k)} \left( \frac{\tilde{x}_2^{(k)} - \tilde{d}_2^{(k)}}{x_2 - \tilde{d}_2^{(k)}} \right)^3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \gamma_n^{(k)} \left( \frac{\tilde{x}_n^{(k)} - \tilde{d}_n^{(k)}}{x_n - \tilde{d}_n^{(k)}} \right)^3 \end{pmatrix} \quad (3.38)$$

*In view of (3.38), it remains to show that the Hessian matrix on the right-hand side of (3.38) is positive semidefinite for all  $x \in \mathcal{I}^{(k)}$ . Since  $\gamma^{(k)}$  is nonnegative (as can be seen in (3.29)), and the two terms  $\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}$  and  $x_j - \tilde{d}_j^{(k)}$  have the same sign in the interval  $\mathcal{I}_j^{(k)}$ , and for all  $x$  in  $\mathcal{I}^{(k)}$ ,  $x^T \nabla^2 \tilde{f}_w^{(k)}(x) x \geq 0$  then  $\tilde{f}_w^{(k)}$  is a convex function on  $\mathcal{I}^{(k)}$ . Furthermore, the function  $\tilde{f}_w^{(k)}$  being well definite and continuous in  $\mathcal{I}^{(k)}$ , this implies the existence of a minimum, which by convexity is the unique critical point  $\tilde{x}_*^{(k)}$ . Now, looking for  $\nabla \tilde{f}_w^{(k)}(x) = 0_{\mathbb{R}^n}$*

we conclude that the optimum  $\tilde{x}_*^{(k)}$  is one solution of the system

$$\begin{pmatrix} f_{,1}(\tilde{x}^{(k)}) + \frac{1}{2}\gamma_1^{(k)}(\tilde{x}_1^{(k)} - \tilde{d}_1^{(k)}) \left(1 - \frac{(\tilde{x}_1^{(k)} - \tilde{d}_1^{(k)})^2}{(x_1 - \tilde{d}_1^{(k)})^2}\right) \\ \vdots \\ f_{,j}(\tilde{x}^{(k)}) + \frac{1}{2}\gamma_j^{(k)}(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}) \left(1 - \frac{(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})^2}{(x_j - \tilde{d}_j^{(k)})^2}\right) \\ \vdots \\ f_{,n}(\tilde{x}^{(k)}) + \frac{1}{2}\gamma_n^{(k)}(\tilde{x}_n^{(k)} - \tilde{d}_n^{(k)}) \left(1 - \frac{(\tilde{x}_n^{(k)} - \tilde{d}_n^{(k)})^2}{(x_n - \tilde{d}_n^{(k)})^2}\right) \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad (3.39)$$

or, equivalently,

$$f_{,j}(\tilde{x}^{(k)}) + \frac{1}{2}\gamma_j^{(k)}(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}) \left(1 - \frac{(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})^2}{(x_j - \tilde{d}_j^{(k)})^2}\right) = 0, \quad j = 1, \dots, n, \quad (3.40)$$

which, after trivial calculations, implies

$$\left(\frac{\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}}{x_j - \tilde{d}_j^{(k)}}\right)^2 = 1 + \frac{2f_{,j}(\tilde{x}^{(k)})}{\gamma_j^{(k)}(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})}. \quad (3.41)$$

Now using (3.30), we can write

$$\frac{2f_{,j}(\tilde{x}^{(k)})}{\gamma_j^{(k)}(\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})} = -\frac{1}{\tilde{\alpha}_j^{(k)}}. \quad (3.42)$$

Therefore, after some simplification, equation (3.41) becomes

$$\left(\frac{\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)}}{x_j - \tilde{d}_j^{(k)}}\right)^2 = 1 - \frac{1}{\tilde{\alpha}_j^{(k)}}. \quad (3.43)$$

By condition (3.31) imposed on the parameter  $\alpha^{(k)}$ , it can be deduced from (3.43) that the solvability of our subproblem can always be guaranteed. Indeed, under this condition, the required identity (3.35) immediately follows from (3.43) and the above mentioned fact that  $\tilde{x}_j^k - \tilde{d}_j^{(k)}$  and  $x_j - \tilde{d}_j^{(k)}$  have the same sign in the interval  $\mathcal{I}_j^{(k)}$ . This completes the proof of the theorem.

### 3.3.1 Convergence study

In this Section, we give our second main result of this chapter, that is sufficient conditions on the data (the point  $\tilde{x}^{(0)}$ , the gradient  $\nabla f$  in a neighborhood of  $\tilde{x}^{(0)}$ , the family  $\text{diag}(H_f(\tilde{x}^{(k)})), k \geq 0$ ), which guarantee that first derivative of  $f$  vanishes in a neighborhood of  $x^*$ , first, and secondly, the convergence of the method to this zero.

To establish our convergence results, we need the following assumptions. We assume that there exist positive constants  $r, M$  and  $\xi < 1$  such that the following assumptions hold: Here where  $\|\cdot\|$  is the standard Euclidean norm on  $\mathbb{R}^n$ .

**Assumption 6**

$$B_r := \left\{ x \in \mathbb{R}^n : \|x - \tilde{x}^{(0)}\| \leq r \right\} \subset \Omega.$$

**Assumption 7**

$$0 < \frac{\tilde{\alpha}_j^{(k)}}{\tilde{\alpha}_j^{(k)} - 1} \leq \frac{M}{2} \gamma_j^k, \quad (k > 0), \quad j = 1, \dots, n.$$

**Assumption 8**

$$\sup_{k>0} \sup_{x \in B} \left\| \nabla f_{,j}(x) - \frac{f_{,j}(x^{(k-1)})}{x_j^{(k-1)} - x_j^{(k)}} e^{(j)} \right\| \leq \frac{\xi}{M},$$

where  $e^{(j)}$  is the vector of  $\mathbb{R}^n$  with  $j$ -th component equal to 1 and all other components equal to 0.

**Assumption 9**

$$0 < \left| f_{,j}(\tilde{x}^{(0)}) \right| \leq \frac{r}{M\sqrt{n}} (1 - \xi).$$

Assumption 7 enforces the quite natural conditions (3.31). Indeed, if condition 7 holds, then (3.31) is also satisfied. Furthermore, Assumption 8 tells us that the coefficient  $\nabla f_{,j}(\tilde{x}^{(k)})$  does not change too much in a neighborhood of  $\tilde{x}^{(0)}$ , and finally for any  $k$  the functions  $\nabla f_{,j}(\tilde{x}^{(k)})$  and  $\frac{f_{,j}(x^{(k-1)})}{x_j^{(k-1)} - \tilde{x}_j^{(k)}}$  have not to change too much in a neighborhood of  $\tilde{x}^{(k)}$ . Assumption 9 only says that  $\left| f_{,j}(\tilde{x}^{(0)}) \right|$  is small enough and that  $f_{,j}(\tilde{x}^{(0)})$  is non zero.

Throughout this subsection, we assume that Assumptions 6-9 hold. The constants  $r, M$  and  $\xi < 1$  that appear in the subsequent analysis are always the constants from Assumptions 6-9. Our aim is to show that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  defined in Theorem 3.3.1 converges geometrically to a point  $x^*$  in the sense that

$$\|\tilde{x}^{(k)} - x^*\| \leq \frac{\xi^k}{1 - \xi} \|x^{(1)} - x^{(0)}\|.$$

**Theorem 3.3.2** *Assume Assumptions 6-9 hold. Let the assumptions of theorem 4.4.3 be valid and let  $G^{(k)}$  be defined by (3.36). Then the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  given by*

$$\tilde{x}^{(k+1)} = \tilde{d}^{(k)} + G^{(k)} \tag{3.44}$$

*is completely contained in the ball  $B_r$ , and converges to the unique zero of  $\nabla f$  in  $B_r$ .*

We first state some auxiliary lemmas, which will be needed in our investigation.

**Lemma 3.3.3** *Let Assumption 7 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be as defined in Theorem 3.3.2. Then, for any positive integer  $k$  the following inequality holds.*

$$\|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}\| \leq M \|\nabla f(\tilde{x}^{(k-1)})\|. \tag{3.45}$$

**Proof 6** Let us fix a positive integer  $k$ . Using (3.35) and (3.37) we may write

$$\begin{aligned}\tilde{x}_j^{(k)} - \tilde{x}_j^{(k-1)} &= \tilde{d}_j^{(k-1)} + (\tilde{x}_j^{(k-1)} - \tilde{d}_j^{(k-1)})\sqrt{\tilde{s}_j^{(k-1)}} - \tilde{x}_j^{(k-1)} \\ &= (\tilde{x}_j^{(k-1)} - \tilde{d}_j^{(k-1)})\left(\sqrt{\tilde{s}_j^{(k-1)}} - 1\right)\end{aligned}\quad (3.46)$$

Now, from (3.31) we have

$$\tilde{s}_j^{(k-1)} > 1, \quad j = 1, \dots, n, \quad (k \geq 1), \quad (3.47)$$

we then immediately deduce

$$\sqrt{\tilde{s}_j^{(k-1)}} < \tilde{s}_j^{(k-1)}.$$

Therefore, by (3.30), (3.37) and (3.46), we arrive at

$$\left|\tilde{x}_j^{(k)} - \tilde{x}_j^{(k-1)}\right| \leq \left|\tilde{x}_j^{(k-1)} - \tilde{d}_j^{(k-1)}\right| \left(\tilde{s}_j^{(k-1)} - 1\right), \quad (3.48)$$

$$\leq \frac{2\tilde{\alpha}_j^{(k-1)}}{(\tilde{\alpha}_j^{(k-1)} - 1)\gamma_j^{(k-1)}} \left|f_{,j}(\tilde{x}^{(k-1)})\right|. \quad (3.49)$$

Finally, combining Assumption 7 and this last inequality, we get the required inequality (3.45).

In order to prove that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  converges geometrically, we need some further preparatory results.

**Lemma 3.3.4** Let Assumption 8 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be defined as in Theorem 3.3.2. Then, for any positive  $k$  the following inequality holds.

$$\left|f_{,j}(\tilde{x}^{(k)})\right| \leq \frac{\xi}{M} \left|\tilde{x}_j^{(k)} - \tilde{x}_j^{(k-1)}\right|, \quad j = 1, \dots, n. \quad (3.50)$$

**Proof 7** Fix a positive integer  $k$ . Let us define  $\tilde{t}_j^{(k-1)}$  by

$$\tilde{t}_j^{(k-1)} := \frac{\gamma_j^{(k-1)}}{2} (\tilde{x}_j^{(k-1)} - \tilde{d}_j^{(k-1)}) - f_{,j}(\tilde{x}^{(k-1)}), \quad (3.51)$$

and the auxiliary function  $\varphi : B \rightarrow \mathbb{R}$  as follows:

$$\varphi(x) := f_{,j}(x) - \frac{f_{,j}(\tilde{x}^{(k-1)})}{\frac{1}{2}\gamma_j^{(k-1)}(\tilde{x}_j^{(k)} - \tilde{x}_j^{(k-1)})} h_j(x),$$

where

$$h_j(x) := -f_{,j}(\tilde{x}^{(k-1)}) + \frac{\gamma_j^{(k-1)}}{2} (x_j - \tilde{x}_j^{(k)} + \tilde{x}_j^{(k-1)} - \tilde{d}_j^{(k-1)}) - \tilde{t}_j^{(k-1)}.$$

Using (3.51), it is easily checked that  $\varphi$  satisfies

$$\begin{aligned}\varphi(\tilde{x}^{(k-1)}) &= 0, \\ \varphi(\tilde{x}^{(k)}) &= f_{,j}(\tilde{x}^{(k)}).\end{aligned}$$

Also it is easy to see that

$$\nabla\varphi(x) = \nabla f_{:,j}(x) - \frac{f_{:,j}(\tilde{x}^{(k-1)})}{\tilde{x}_j^{(k-1)} - \tilde{x}_j^{(k)}} e^{(j)}. \quad (3.52)$$

Then, from the mean-value theorem and Assumption 8 we get

$$\begin{aligned} \left| f_{:,j}(\tilde{x}^{(k)}) \right| &= \left| \varphi(\tilde{x}^{(k)}) - \varphi(\tilde{x}^{(k-1)}) \right| \\ &\leq \sup_{x \in B_r} \|\nabla\varphi(x)\| \|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}\| \\ &\leq \sup_{x \in B_r} \left\| \nabla f_{:,j}(x) - \frac{f_{:,j}(\tilde{x}^{(k-1)})}{\tilde{x}_j^{(k-1)} - \tilde{x}_j^{(k)}} e^{(j)} \right\| \|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}\| \\ &\leq \frac{\xi}{M} \|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}\|. \end{aligned} \quad (3.53)$$

This shows that the required inequality (3.50) holds true for any positive integer  $k$ .

The next result shows that for all  $k$  the iterate  $\tilde{x}^{(k)}$  remains in the interval  $B_r$ .

**Lemma 3.3.5** *Let Assumption 7-8 be satisfied and let the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  be as defined in Theorem 3.3.2. Assume that the starting point  $\tilde{x}_0$  belongs to the interval  $B_r$ , where  $r$  is defined in Assumption 6. Then, all terms of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  lie inside the ball  $B_r$ .*

**Proof 8** *Indeed, combining inequalities (3.45), (3.50) of Lemmas 3.3.3 and 3.3.4 respectively, we immediately obtain*

$$\|\tilde{x}^{(k)} - \tilde{x}^{(k-1)}\| \leq \xi \|\tilde{x}^{(k-1)} - \tilde{x}^{(k-2)}\| \leq \dots \leq \xi^{k-1} \|\tilde{x}^{(1)} - \tilde{x}^{(0)}\|, \quad (3.54)$$

and therefore we have

$$\begin{aligned} \|\tilde{x}^{(k)} - \tilde{x}^{(0)}\| &\leq \sum_{l=1}^k \|\tilde{x}^{(l)} - \tilde{x}^{(l-1)}\| \\ &\leq \left( \sum_{l=1}^k \xi^{l-1} \right) \|\tilde{x}^{(1)} - \tilde{x}^{(0)}\| \\ &\leq \frac{\|\tilde{x}^{(1)} - \tilde{x}^{(0)}\|}{1 - \xi}. \end{aligned} \quad (3.55)$$

Finally, applying inequality (4.4a) for  $k = 1$  and using Assumption 9 we immediately get

$$\|\tilde{x}^{(k)} - \tilde{x}^{(0)}\| \leq r, \quad (3.56)$$

which shows that each  $\tilde{x}^{(k)}$  belongs to  $B_r$ .

As a consequence of the previous three lemmas, we are now in a position to prove Theorem 3.3.2.

*Proof of Theorem 3.3.2.* Since the entire sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  remains in the (closed) ball  $B_r$  by Lemma 3.3.5, every limit point of this sequence belongs to this set, too. Hence, it remains to show that the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$

converges. To this end, we first note that, for  $k \geq 0$  and  $l \geq 0$ , we have

$$\begin{aligned} \left\| \tilde{x}^{(k+l)} - \tilde{x}^{(k)} \right\| &= \left\| \sum_{v=0}^{l-1} (\tilde{x}^{(k+v+1)} - \tilde{x}^{(k+v)}) \right\| \\ &\leq \sum_{v=0}^{l-1} \left\| \tilde{x}^{(k+v+1)} - \tilde{x}^{(k+v)} \right\| \\ &\leq \xi^k \sum_{v=0}^{l-1} \xi^v \left\| \tilde{x}^{(1)} - \tilde{x}^{(0)} \right\| \\ &\leq \frac{\xi^k}{1-\xi} \left\| \tilde{x}^{(1)} - \tilde{x}^{(0)} \right\|, \end{aligned} \quad (3.57)$$

then the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  is a Cauchy sequence. Being Cauchy in  $B_r$ , it has a limit,  $x^*$ , in  $B_r$ . Now, thanks to the continuity of  $f_j$  on  $B_r$  and the convergence of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  imply

$$|f_j(\tilde{x}^*)| = \lim_{k \rightarrow +\infty} |f_j(\tilde{x}^{(k)})| \leq \frac{\xi}{M} \lim_{k \rightarrow +\infty} |\tilde{x}_j^{(k)} - \tilde{x}_j^{(k-1)}| = 0, \quad j = 1, \dots, n, \quad (3.58)$$

and then  $\nabla f(\tilde{x}^*) = 0$ . Passing to the limit for  $l$  tending to  $\infty$  in (3.57), we deduce that

$$\left\| \tilde{x}^{(k)} - \tilde{x}^* \right\| \leq \frac{\xi^k}{1-\xi} \left\| \tilde{x}^{(1)} - \tilde{x}^{(0)} \right\|, \quad (3.59)$$

which shows the geometric convergence of the sequence  $\{\tilde{x}^{(k)}\}_{k \geq 0}$  to  $\tilde{x}^*$ .

We are now in a position to prove that  $\nabla f$  has an unique zero in  $B_r$ . To this end, we proceed by contradiction, assuming that  $\nabla f$  has another zero  $\tilde{y}^* \in B_r$ . Let us introduce the auxiliary function

$$\lambda_j(x) = \frac{\tilde{x}_j^{(1)} - \tilde{x}_j^{(0)}}{f_{j,j}(\tilde{x}_j^{(0)})} \left( f_{j,j}(x) - \frac{f_{j,j}(\tilde{x}_j^{(0)})}{\tilde{x}_j^{(0)} - \tilde{x}_j^{(1)}} (x_j - \tilde{x}_j^*) \right),$$

which satisfies  $\lambda_j(\tilde{x}^*) = 0$  and  $\lambda_j(\tilde{y}^*) = \tilde{y}_j^* - \tilde{x}_j^*$ . Therefore, applying (3.45) for  $k = 1$ , it follows from the mean value theorem and Lemma 3.3.3, inequality (3.45) for  $k = 1$ ,

$$\begin{aligned} |\tilde{x}_j^* - \tilde{y}_j^*| &= |\lambda_j(\tilde{x}^*) - \lambda_j(\tilde{y}^*)| \\ &\leq \sup_{x \in B_r} \left\| \nabla \lambda_j(x) \right\| \|\tilde{x}^* - \tilde{y}^*\| \\ &\leq \left\| \frac{\tilde{x}_j^{(1)} - \tilde{x}_j^{(0)}}{f_{j,j}(\tilde{x}_j^{(0)})} \right\| \sup_{x \in B} \left\| \nabla f_{j,j}(x) - \frac{f_{j,j}(\tilde{x}_j^{(0)})}{\tilde{x}_j^{(0)} - \tilde{x}_j^{(1)}} e^{(j)} \right\| \|\tilde{x}^* - \tilde{y}^*\| \\ &\leq M \frac{\xi}{M} \|\tilde{x}^* - \tilde{y}^*\| \\ &= \xi \|\tilde{x}^* - \tilde{y}^*\|. \end{aligned} \quad (3.60)$$

This yields  $\tilde{x}^* = \tilde{y}^*$  since  $\xi < 1$ . Thus, the theorem is proved.

### 3.3.2 Choice of the parameters $\tilde{\alpha}^{(k)}$

Here we give certain details, for the choice of the moving asymptotes  $\tilde{d}^{(k)}$ , which are in general updated in each iteration. The introduction of the terms  $\tilde{\alpha}^{(k)}$  in the moving asymptote  $\tilde{d}^{(k)}$ , as defined in (3.30) and (3.31)

respectively, is the basic difference between the [3] method and its variant presented here. These coefficients are essentially tuning parameters for global convergence for the present method, and of course their choices are a crucial point for the behaviour of the algorithms discussed in this chapter. To establish our convergence theorem, we have imposed that the parameter  $\tilde{\alpha}^{(k)}$  must satisfy Assumption 7. By this latter, after some manipulation, it can be deduced that

$$\tilde{\alpha}_j^{(k)} \geq 1 + \frac{2}{M\gamma_j^{(k)}}, \quad j = 1, \dots, n, \quad (3.61)$$

then we can choose the parameters  $\tilde{\alpha}_j^{(k)} \in \left[1 + \frac{2}{M\gamma_j^{(k)}}, +\infty\right]$ ,  $j = 1, \dots, n$ . Note that this choice obviously guarantees that  $\tilde{\alpha}^{(k)} > 1$ , which is one important assumption in Theorems 3.3.1 and 3.3.2 for the solvability of the subproblems. Although this is not an explicit choice, it was found to work well in practice by taking a suitable value for the parameter  $M$ .

### 3.3.3 Description of algorithm

The results of the previous section may be used to construct the following algorithm.

---

#### Algorithm 2 Modified Method of Moving Asymptotes

---

- 1: Input:  $\tilde{x}^{(0)} \in \mathbb{R}^n$ ,  $w, M_1, M_2 \geq 1$ , (and an optional error tolerance  $\varepsilon > 0$ ).
  - 2:  $k = 0$
  - 3: REPEAT
  - 4: For  $j = 0, 1, \dots, n$
  - 5:  $\gamma_j^{(k)} = \left| f_{,jj}(\tilde{x}^{(k)}) + w(\tilde{x}^{(k)})f_{,j}(\tilde{x}^{(k)}) \right|$ ,
  - 6:  $\tilde{\alpha}_j^{(k)} = M_1 \left( 1 + \frac{2}{M_2\gamma_j^{(k)}} \right)$ ,
  - 7:  $\tilde{d}_j^{(k)} = \tilde{x}_j^{(k)} + 2\tilde{\alpha}_j^{(k)} \frac{f_{,j}(\tilde{x}^{(k)})}{\gamma_j^{(k)}}$ ,
  - 8:  $\tilde{s}_j^{(k)} = \frac{\tilde{\alpha}_j^{(k)}}{\tilde{\alpha}_j^{(k)} - 1}$ ,
  - 9:  $\tilde{x}_j^{(k+1)} = \tilde{d}_j^{(k)} + (\tilde{x}_j^{(k)} - \tilde{d}_j^{(k)})\sqrt{\tilde{s}_j^{(k)}}$ ,
  - 10: while  $\left\| \nabla f(\tilde{x}^{(k)}) \right\| > \varepsilon$ .
  - 11:  $k \leftarrow k + 1$
- 

## 3.4 Minimizing Non-convex Non-Separable Functions

The separability is a measure of difficulty of different objective functions. In general the separable functions are relatively easy to solve, when compared with their inseparable counterpart, because each variable of a separable function is independent of the other variables. If all the parameters or variables are independent, then a sequence



of  $n$  independent optimization processes can be performed. As a result, each design variable or parameter can be optimized independently, and if the objective function  $f$  is separable, then, for  $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$

$$\arg \min_x f(x) = \arg \min_{x_1, x_2, \dots, x_n} f(x_1, x_2, \dots, x_n) \quad (3.62)$$

$$= \arg \min_{x_1, x_2, \dots, x_n} \sum_{i=1}^n f_i(x_i) \quad (3.63)$$

$$= \sum_{i=1}^n \arg \min_{x_i} f_i(x_i) \quad (3.64)$$

where  $f(x) = \sum_{i=1}^n f_i(x_i)$ . On the other hand, a function is called non separable, if its variables show inter-relation among themselves or are not independent. If the objective function variables are independent of each other, then the objective functions can be decomposed into sub-objective functions. Then, each of these sub-objectives involves only one decision variable, and Algorithm 2 solves the problem (3.1) for each variable independently of others, and then is like we minimize  $n$  functions of dimension 1. According to [40, 59], the general condition of separability to see if the function is easy to optimize or not is given as

$$\frac{\partial f}{\partial x_i}(\bar{x}) = g(x_i)h(\bar{x}). \quad (3.65)$$

The dimensionality is one of the difficulty for solving the problem (3.1), this difficulty increases with the problem's dimension. According to [77, 78, 16, 79] the search space increases exponentially with the number of parameters or dimension. For highly nonlinear problems, this dimensionality may be a significant barrier for almost all optimization algorithms. In this section, we extend Algorithm 2 for the non separable non convex optimization problem, which consists of a cyclic update of the variables  $x_i$ , let's consider the problem (3.1), where here the objective function is non separable: we can formulate this problem as: At each iteration of this method, the function is minimized with respect to a single of variables while the rest of the variables are held fixed. More specifically, at iteration  $k$  of the algorithm, the variable  $x_i$  is updated by solving the following subproblems

$$\left\{ \begin{array}{l} x_{\sigma(1)}^{(k+1)} := \arg \min_{x_{\sigma(1)}} f(x_{\sigma(1)}, x_{\sigma(2)}^{(k)}, \dots, x_{\sigma(i)}^{(k)}, \dots, x_{\sigma(n)}^{(k)}) \\ x_{\sigma(2)}^{(k+1)} := \arg \min_{x_{\sigma(2)}} f(x_{\sigma(1)}^{(k)}, x_{\sigma(2)}, \dots, x_{\sigma(i)}^{(k)}, \dots, x_{\sigma(n)}^{(k)}) \\ \vdots \\ x_{\sigma(i)}^{(k+1)} := \arg \min_{x_{\sigma(i)}} f(x_{\sigma(1)}^{(k+1)}, x_{\sigma(2)}^{(k+1)}, \dots, x_{\sigma(i-1)}^{(k+1)}, x_{\sigma(i)}, \dots, x_{\sigma(n)}^{(k)}) \\ \vdots \\ x_{\sigma(n)}^{(k+1)} := \arg \min_{x_{\sigma(n)}} f(x_{\sigma(1)}^{(k+1)}, x_{\sigma(2)}^{(k+1)}, \dots, x_{\sigma(i)}^{(k+1)}, \dots, x_{\sigma(n-1)}^{(k+1)}, x_{\sigma(n)}), \end{array} \right. \quad (3.66)$$

where  $\sigma$  is a uniformly random permutation of  $\{1, \dots, n\}$ . Let us use  $x^{(k)}$  to denote the sequence of iterates generated by this algorithm, where  $x^{(k)} \triangleq (x_{\sigma(1)}^{(k)}, x_{\sigma(2)}^{(k)}, \dots, x_{\sigma(n)}^{(k)})$  at iteration  $k$ , the selected variable (say variable  $i$ ) is computed by solving the  $n$  following subproblems

$$\left\{ \begin{array}{l} \arg \min f_{\sigma(i)}(x_{\sigma(i)}) \\ \text{s.t. } x_{\sigma(i)} \in \mathbb{R} \end{array} \right. \quad (3.67)$$

where  $f_{\sigma(i)}(x_{\sigma(i)}) := f(x_{\sigma(1)}^{(k+1)}, x_{\sigma(2)}^{(k+1)}, \dots, x_{\sigma(i-1)}^{(k+1)}, x_{\sigma(i)}, \dots, x_{\sigma(n)}^{(k)})$  is an approximation of the original objective function at the point  $x^{(k-1)}$ . Algorithm 3 summarizes the process of solving a non convex optimization problem for a non separable objective function:

---

**Algorithm 3** Modified version for non separable function
 

---

- 1: Input a feasible point  $x^{(0)}$ , and set  $k = 0$
- 2: **Repeat**
- 3:  $k \leftarrow k + 1$ , choose a permutation  $\sigma$ ,
- 4: If  $\sigma := \{1, \dots, n\}$
- 5: For  $i = 1, \dots, n$
- 6: Solve by the **algorithm 2** the  $n$  one-dimensional problems

$$\begin{cases} \arg \min f_i(x_i) \\ \text{s.t. } x_i \in \mathbb{R} \end{cases} \quad i = 1, \dots, n$$

$$7: x^{(k+1)} = \left( \arg \min_{x_1} f_1(x_1), \dots, \arg \min_{x_n} f_n(x_n) \right)$$

- 8: **Until** some convergence criterion is met
- 

Note that Algorithm 3 reduces to Algorithm 2 for the case where condition of separability (3.65) is valid, and then all of the variables are independent of each other.

### 3.5 Numerical examples

We employ the present method (designated as present) to solve some nonlinear, non-convex optimization problems and compare it with the [3] method, Newton's method and the BFGS method using two kinds of weight functions and four test functions:

$$f_1(x) = -\left(\frac{1}{3}x^3 + \frac{5}{2}x^2 + 3x - \exp(x)\right),$$

$$f_2(x, y) = \frac{1}{4}(x^4 + (y-1)^4) + \frac{4}{3}x^3 - 15\left(x + \frac{2}{15}y\right) + 3,$$

$$f_3(x, y) = -\left(\exp(x) + \exp(2y) + \frac{(x^3 + y^3)}{3} - (x^2 + y^2 + 3(x+y) + 12)\right).$$

$$f_4(x, y, z) = \frac{1}{2}\left(\exp(x^2) + 2\exp(y) + \frac{1}{2}(z-3)^4\right) + 3\left(\sin(x) - \frac{1}{6}\sin(2x)\right) - \left(\frac{1}{3}y^3 + \frac{5}{2}y^2 + 3(y+z) - 6\right)$$

It is assumed that all methods use the finite difference method to compute first and second derivatives.

Numerical results are summarized in Tables 3.1 - 3.5, where for each weight function  $w$ , we present the objective functions, the starting points, the methods used, the number of iterations  $N$  to obtain the objective value of the obtained optimal solution  $x^{(N)}$  and  $f(x)$  at  $x^{(N)}$ . We use the following stopping criteria  $\left\|\nabla f(\tilde{x}^{(N)})\right\| \leq \varepsilon$ , (the absolute value of the derivative of the function is less than or equal to the tolerance). For numerical illustrations we used different values of  $\varepsilon$ . Therefore, when the stopping criterion is satisfied,  $\tilde{x}_* = \tilde{x}^{(N)}$  is taken as the

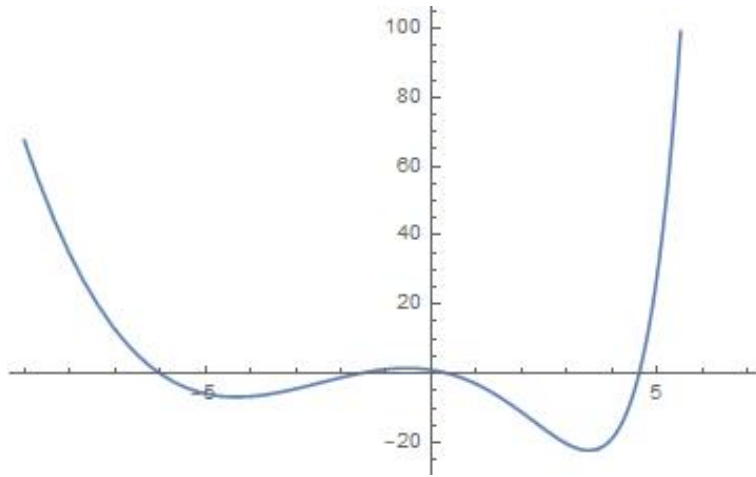
optimal solution. In Tables 3.1 - 3.5 div. means that the stopping criteria is not satisfied.

The test results in Tables 3.1 and 3.2 show that for all of the functions we tested, the present method is better than the [3] method, Newton's method and the BFGS Method. It always converges even if the starting point is very far from the true solution, and it never requires more iterations than the other three methods. Finally we must mention that in many other test cases in some examples the present method may converge when these latter diverge simultaneously. Many of them have been shown in this section. These characteristics give a strong advantage over these other methods.

we consider the function:

$$f_1 : \mathbb{R} \longrightarrow \mathbb{R}$$

$$x \longrightarrow -\left(\frac{1}{3}x^3 + \frac{5}{2}x^2 + 3x - \exp(x)\right),$$



Initial point	Method	$N$	$x_N$	$f(x_N) \simeq \min f(x)$	$\epsilon$
$\tilde{x}^{(0)} = -2.5$	.	.	.	.	$10^{-16}$
.	Newton	div.	.	.	.
.	BFGS	div.	.	.	.
.	[3]	div.	.	.	.
.	Present	20	-4.306510	-6.809174	.
$\tilde{x}^{(0)} = 12$	.	.	.	.	$10^{-15}$
.	Newton	17	.	.	.
.	BFGS	17	.	.	.
.	[3]	31	3.482467	-2.230440e+001	.
.	Present	12	.	.	.

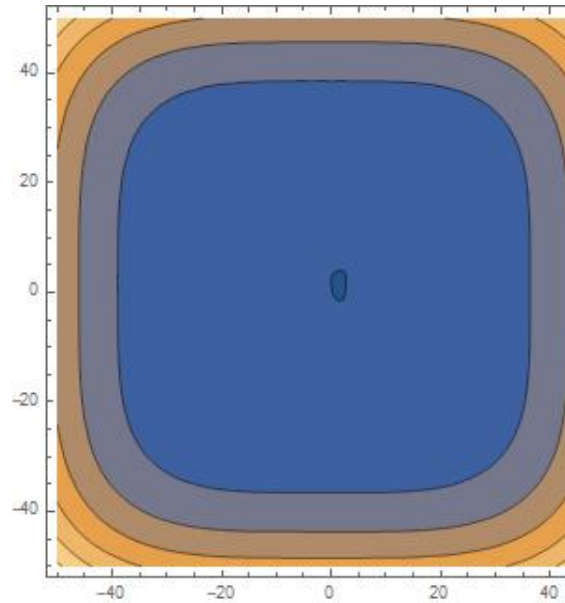
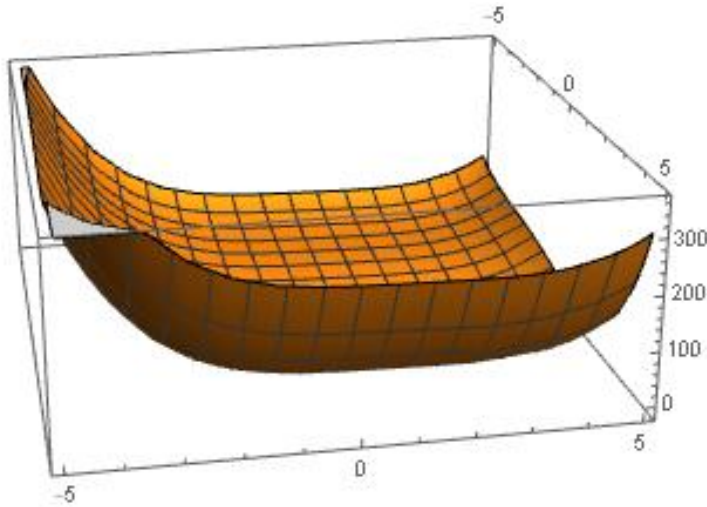
Table 3.1 Numerical comparison of the [3] method, Newton's method, the BFGS Method and the present method. Here  $w(x) = (1 + |x|)^{1/2} \exp(-2|x|)$  and  $\tilde{\alpha}^{(k)} = 2 \left(1 + \frac{1}{4\tilde{c}^{(k)}}\right)$

Initial point	$f(x_0)$	Method	$N$	$x_N$	$f(x_N) \simeq \min f(x)$	$\epsilon$
$\tilde{x}^{(0)} = -30 * 10^{100}$	9.00e+303	.	.	.	.	$10^{-7}$
.	.	Newton	340	.	.	
.	.	BFGS	340	.	.	
.	.	[3]	div.	.	.	
.	.	Present	238	-4.306510	-6.809174	
$\tilde{x}^{(0)} = -21 * 10^{50}$	30.87e+152	.	.	.	.	$10^{-12}$
.	.	Newton	173	.	.	
.	.	BFGS	173	.	.	
.	.	[3]	div.	.	.	
.	.	Present	127	-4.306510	-6.809174	
$\tilde{x}^{(0)} = 30 * 10^{10}$	8.99e+033	.	.	.	.	$10^{-15}$
.	.	Newton	44	.	.	
.	.	BFGS	44	.	.	
.	.	[3]	div.	3.482467	-2.230440e + 001	
.	.	Present	40	.	.	

Table 3.2 Numerical comparisons of the [3] method, Newton’s method, the BFGS Method and the present paper. Here  $w(x) = (1 + |x|)^{-4} \exp(-10 |x|^{0.5}) \log(e + |x|)^{10}$  and  $\tilde{\alpha}^{(k)} = 3 \left( 1 + \frac{1}{10\tilde{c}^{(k)}} \right)$ .

$$f_2 : \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$$

$$(x, y) \rightarrow \frac{1}{4} (x^4 + (y - 1)^4) + \frac{4}{3} x^3 - 15 \left( x + \frac{2}{15} y \right) + 3,$$

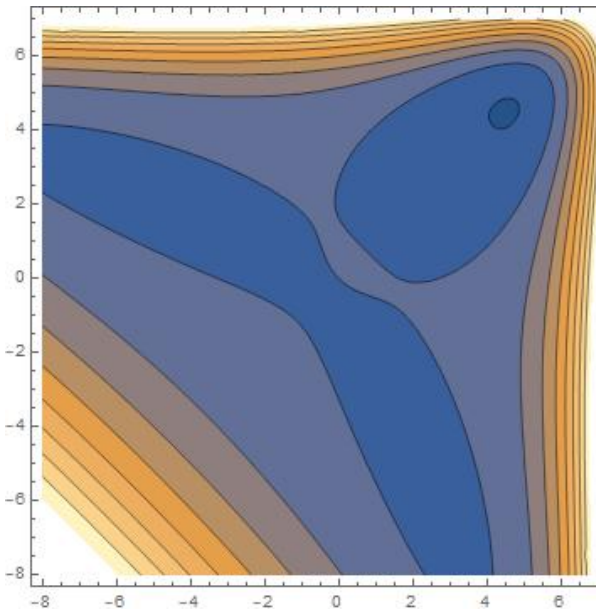
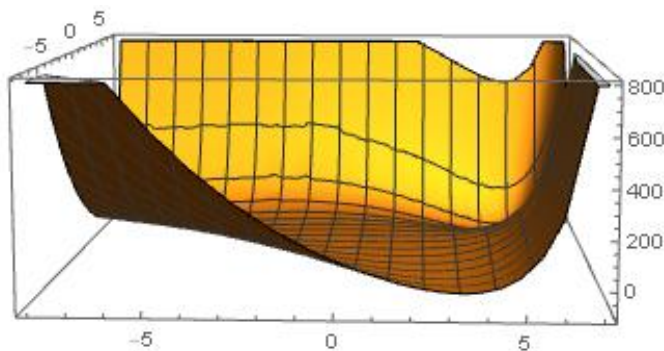


Initial point	Method	$N$	$x_N$	$f(x_N) \simeq \min f(x)$	$\epsilon$
$\tilde{x}^{(0)} = (1, -1)$	.	.	.	.	$10^{-16}$
.	Newton	div.	.	.	.
.	BFGS	div.	$\begin{pmatrix} 1.63198080556606e+000 \\ 1.79370052598410e+000 \end{pmatrix}^T$	$-17.3991210151531e+000$	.
.	[3]	13.	.	.	.
.	Present	5	.	.	.
$\tilde{x}^{(0)} = (0, 0)$	.	.	.	.	$10^{-15}$
.	Newton	8	$\begin{pmatrix} 1.63198080556606e+000 \\ 1.79370052598410e+000 \end{pmatrix}^T$	$-17.3991210151531e+000$	.
.	BFGS	7	.	.	.
.	[3]	6	.	.	.
.	Present	4	.	.	.

Table 3.3 Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method. Here  $w(x) = (1 + |x|)^{1/2} \exp(-2|x|)$  and  $\tilde{\alpha}^{(k)} = \left( 2 \left( 1 + \frac{1}{4\gamma_1^{(k)}} \right), 4 \left( 1 + \frac{1}{3\gamma_2^{(k)}} \right) \right)$ .

$$f_3: \mathbb{R} \times \mathbb{R} \longrightarrow \mathbb{R}$$

$$(x, y) \longrightarrow - \left( \exp(x) + \exp(2y) + \frac{(x^3 + y^3)}{3} - (x^2 + y^2 + 3(x + y) + 12) \right),$$



Initial point	Method	$N$	$X_N = (x_N, y_N)$	$f(X_N) \simeq \min f(x)$	$\epsilon$
$\tilde{x}^{(0)} = (15, 10)$	.	.	.	.	$10^{-16}$
.	Newton	div.	.	.	.
.	BFGS	div.	$\begin{pmatrix} -895.108649662366e-003 \\ -918.740159643646e-003 \end{pmatrix}^T$	8.13353253920338e+000	.
.	[3]	17.	.	.	.
.	Present	10	.	.	.
$\tilde{x}^{(0)} = (0, 0)$	.	.	.	.	$10^{-15}$
.	Newton	8	.	.	.
.	BFGS	7	$\begin{pmatrix} -895.108649662366e-003 \\ -918.740159643646e-003 \end{pmatrix}^T$	8.13353253920338e+000	.
.	[3]	6	.	.	.
.	Present	4	.	.	.

Table 3.4 Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the

present method. Here  $w(x) = (1 + |x|)^{1/2} \exp(-2|x|)$  and  $\tilde{\alpha}^{(k)} = \left( 2 \left( 1 + \frac{1}{5\gamma_1^{(k)}} \right), 3 \left( 1 + \frac{1}{10\gamma_2^{(k)}} \right) \right)$ .

$$f_4: \quad \mathbb{R}^3 \quad \longrightarrow \quad \mathbb{R}$$

$$(x, y, z) \quad \longrightarrow \quad \frac{1}{2} \left( \exp(x^2) + 2 \exp(y) + \frac{1}{2} (z-3)^4 \right) + 3 \left( \sin(x) - \frac{1}{6} \sin(2x) \right) - \left( \frac{1}{3} y^3 + \frac{5}{2} y^2 + 3(y+z) - 6 \right)$$

Initial point	Method	$N$	$X_N = (x_N, y_N, z_N)$	$f(X_N) \simeq \min f(x)$	$\epsilon$
$\tilde{x}^{(0)} = (10, 100, 200)$	.	.	.	.	$10^{-16}$
.	Newton	div.	.	.	.
.	BFGS	div.	$\begin{pmatrix} -957.55090028105e-003 \\ 3.48246759967065e+000 \\ 2.44224957030741e+000 \end{pmatrix}^T$	-23.9955009815414e+000	.
.	[3]	div.	.	.	.
.	Present	110	.	.	.
$\tilde{x}^{(0)} = (2, 5, 3)$	.	.	.	.	$10^{-15}$
.	Newton	19	.	.	.
.	BFGS	19	$\begin{pmatrix} -957.55090028105e-003 \\ 3.48246759967065e+000 \\ 2.44224957030741e+000 \end{pmatrix}^T$	-23.9955009815414e+000	.
.	[3]	22	.	.	.
.	Present	12	.	.	.

Table 3.5 Numerical comparisons of the [3] method, Newton's method, the BFGS Method and the present method in three dimension.

$$\text{Here } w(x) = (1 + |x|)^{1/4} \exp(-20|x|) \text{ and } \tilde{\alpha}^{(k)} = \left( 5 \left( 1 + \frac{1}{7\gamma_1^{(k)}} \right), 2 \left( 1 + \frac{1}{4\gamma_2^{(k)}} \right), 4 \left( 1 + \frac{1}{3\gamma_3^{(k)}} \right) \right).$$

The process described in section 3.4 has been implemented in *Matlab*. The algorithm was terminated when the norm of the gradient of  $f$   $\|\nabla f(x^{(k)})\|$  was less than a specified tolerance  $\epsilon$ . For all of test functions, the tolerance  $\epsilon$  was taken to be very close to 0, in order to find the most exact solution. For Wood's and Powell's

functions, the  $\varepsilon$  was set  $10^{-25}$  and for Branin's function  $\varepsilon = 10^{-20}$ .

For all the test functions in the table 3.6, the results include the number of variable of the objective function, respectively the number of iterations and evaluations functions  $N_{iter}$  and  $N_{eval}$  required to achieve convergence and the final function value, and the CPU time (s) of the program for each test problem.

Functions	Number of variables	$N_{iter}$	$N_{eval}$	$f(X_N) \simeq \min f(x)$	CPU time (s)
Wood's function	2	7	28	$452.7579 * 10^{-30}$	0.004
	3	12	35	$-8.776523408$	0.025
	4	15	51	$12.7068 * 10^{-30}$	0.067
Powell's function	2	13	42	$8.54987 * 10^{-18}$	0.003
	3	18	60	$36.0706 * 10^{-21}$	0.006
	4	34	73	$14.3158 * 10^{-24}$	2.077
Branin's function	2	30	120	$397.887357 * 10^{-3}$	0.016

Table 3.6 Numerical result for non separable functions

**Remark 3.5.1** *it is easy to check that the values found in the tables correspond to the minima and minimizer of each test function by calculating the gradients of each function at the point  $x_N$ , conclude that it's near to zero, and check that the Hessian matrix of this function at this point is positive definite.*

### 3.6 Conclusion

The proposed modified method of moving asymptotes has been computationally shown that it needs only a small number of iterations to converge to the exact solution up to the specified error tolerance. The algorithm is easy to use since all tuning parameters are automatically chosen. Furthermore, test examples on nonconvex and non-separable functions confirm that the algorithm is expected to be more efficient than the [3] method, Newton's method and the BFGS method. It also has the advantage that, under appropriate conditions, global convergence of the algorithm is guaranteed.

## Chapter 4

# NEW APPROACH OF MOVING ASYMPTOTES METHOD USING A SPECTRAL PARAMETERS FOR BOUND CONSTRAINED OPTIMIZATION

### 4.1 Introduction

In this chapter we present a new extension of the method of moving asymptotes for solving nonlinear non convex optimization problems, that use information from the current and previous iteration and an asymptotes serves to accelerate and stabilize the process convergence in the neighbourhood of the exact solution of the initial problem.

We consider the bound constrained optimization problem

$$\begin{aligned} \min f(x) \\ \text{s.t. } x \in \Omega = \{x \in R^n | l \leq x \leq u\} \end{aligned} \quad (4.1)$$

where  $l = (l^1, l^2, \dots, l^n)^T, u = (u^1, u^2, \dots, u^n)^T$  with  $-\infty \leq l^i < u^i \leq +\infty$  for  $i = 1, 2, \dots, n$ . We denote by  $g(x) = (g^1(x), g^2(x), \dots, g^n(x))^T$  the gradient of  $f$  at  $x$ .

We say that a vector  $\bar{x} \in \Omega$  is a stationary point of problem 4.1 if it satisfies

$$\begin{cases} \bar{x}^i = l^i & \Rightarrow g^i > 0 \\ l^i < \bar{x}^i < u^i & \Rightarrow g^i = 0 \\ \bar{x}^i = u^i & \Rightarrow g^i < 0 \end{cases} \quad (4.2)$$

Problem (4.1) is very important in practical optimization, and many practical problems can be converted into (4.1). In addition, problem (4.1) is often a subproblem of augmented Lagrangian or penalty schemes for



general constrained optimization. Hence it has received much attention in recent decades, and many numerical algorithms have been developed. A popular one among these methods is the method of moving asymptotes which was originally proposed in [69] and extended in [72, 70, 71, 73]. the advantage of this method is that it is quite easy to implement, very effective for large scale problems and converge to the exact solution from any start point. and a second important advantage is keeping the feasibility of the iterate and thanks to the separability of the approximations the process of its algorithm is not expensive.

Moreover, our method is considered more fast than the optimal gradient method for unconstrained optimization.

To speed up the convergence of the classical method of moving asymptotes we propose in this paper an explicit scheme for solving the kind of problems (4.1) which can be seen as an extension of the method of moving asymptotes proposed by Svanberg [72] for unconstrained optimization.

The method of moving asymptotes was originally proposed in [69] for convex objective functions and further analyzed in [72, 70, 71, 73] for the choice of the method's parameters and a modified approximations to achieve a good results.

Recently, by replacing the second order information in the expression of the asymptotes by a spectral parameters in [31, 32] defined a new iterate scheme and proposed a new update of these asymptotes for constrained optimization. The new method is finitely convergent for the convex functions and globally convergent for the general functions based on the updating of the asymptotes with respect to a spectral parameters approximate the second derivative.

The numerical results show that the new approach of the moving asymptotes method works better than the classical method proposed in [8, 9] and more easy to implemented thanks to the clarity of the instructions of the algorithm generated by this approach. In this paper, we aim to extend the Method of moving asymptotes to be independent with keeping the main properties of this method, the separability, the order of approximations and the accuracy of the solutions found by the generated algorithm.

The chapter is organized as follows. In section 2, we introduce the univariate approximations. In section 3 we present the method in the multidimensional case and establish the explicit scheme of these approximations. In section 4 we propose the choice of the parameters of our news approximation functions, these choice give a good convergence to the original solution and control the convergence of the process and a numerical results are given in Section 5.

## 4.2 Univariate objective function

### 4.2.1 Position of the problem and approximations

We start by studying the one-dimensional nonlinear programming problem

$$\begin{aligned} & \text{minimize} && f(x) \\ & \text{subject to} && l \leq x \leq u, \end{aligned} \tag{4.3}$$

where  $x \in IR$  is a variable,  $l$  and  $u$  are given real numbers such that  $l < u$ ,  $f$  are real-valued typically twice continuously differentiable function of  $x$ ,

Since the simplicity of the one-dimensional case allows to detail all the necessary steps with very simple computations, let us first consider the general optimization problem (4.3) of a single real variable. To this end, we first list the necessary notation and terminology.

Let  $n := 1$  and  $\Omega \subset IR$  be an open subset and  $f : \Omega \mapsto IR$  be a given twice differentiable function in  $\Omega$ . Throughout, we assume that  $f'$  does not vanish at a given suitable initial point  $x^{(0)} \in \Omega$ , that is  $f'(x^{(0)}) \neq 0$ , since if this is not the case we have nothing to solve. Starting from the initial design point  $x^{(0)}$  the iterates  $x^{(k)}$  are computed successively by solving sub-problems of the form: Find  $x^{(k+1)}$  such that

$$f(x^{(k+1)}) = \min_{x \in \Omega} f^{(k)}(x), \quad (4.4)$$

where the approximating function  $\tilde{f}^{(k)}$  of the objective function  $f$  at the  $k$ -th externe iteration and  $r$ -th inner iteration has the following form:

$$\begin{aligned} f^{(k)}(x) &= a^{(k)} + b^{(k)} \left( \frac{1}{x^{(k)} - d^{(k)}} - \frac{1}{x - d^{(k)}} \right) (x^{(k)} - d^{(k)})^2 \\ &+ \eta^{(k)} \left( \frac{1}{2} \frac{(x^{(k)} - d^{(k)})^3}{x - d^{(k)}} + \frac{1}{2} (x^{(k)} - d^{(k)}) ((x - 2x^{(k)} + d^{(k)})) \right) \end{aligned} \quad (4.5)$$

$$\eta^{(k)} = \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T s_{k-1}} > 0, \quad (4.6)$$

where  $s_{k-1} = x^{(k)} - x^{(k-1)}$  and  $y_{k-1} = f'(x^{(k)}) - f'(x^{(k-1)})$ . The parameters  $a^{(k)}$ ,  $b^{(k)}$  are adjusted such that a first order approximation is satisfied, i.e

$$f^{(k)}(x^{(k)}) = f(x^{(k)}) \quad (4.7)$$

$$(f^{(k)})'(x^{(k)}) = f'(x^{(k)}) \quad (4.8)$$

$\eta^{(k)}$  is the spectral parameter, and

$$d^{(k)} = x^{(k)} + \sigma^{(k)}, \quad (4.9)$$

$$S^{(k)} := \begin{cases} -1 & \text{if } g(x^{(k)}) < 0, \\ +1 & \text{if } g(x^{(k)}) > 0. \end{cases}$$

Where the asymptotes  $d^{(k)}$  are adjusted heuristically as the optimization progresses or are guided by a proposed given function whose first and second derivative are evaluated at the current iteration point  $x^{(k)}$ . Also, the approximated parameters  $a^{(k)}$ ,  $b^{(k)}$  and  $\eta^{(k)}$  will be determined for each iteration ( $k$ ). To evaluate them, we use the objective function value, its first derivative, as well as its second derivatives at  $x^{(k)}$ . The parameters  $a^{(k)}$ ,  $b^{(k)}$  are determined i such a way that the following set of interpolation conditions are satisfied

$$\begin{aligned} f^{(k)}(x^{(k)}) &= f(x^{(k)}) \\ (f^{(k)})'(x^{(k)}) &= f'(x^{(k)}) \end{aligned}$$

Therefore, by a simple computation, we verify that  $a^{(k)}$ ,  $b^{(k)}$  and  $c^{(k)}$  are explicitly given by:

$$\begin{aligned} a^{(k)} &= f(x^{(k)}) \\ b^{(k)} &= f'(x^{(k)}) \end{aligned}$$

Note that unlike the first order method discussed earlier, here second order version data in addition to data from previous iteration point are used. We now explain how the asymptotes can be chosen. We introduce the following definition: In what follows, we will always assume that the sequence of asymptotes  $\{d^{(k)}\}$  is chosen in our approximate problems such that it forms a sequence of asymptotes. By defining the suitable index set

$$\mathcal{I}^{(k)} = ]-\infty, d^{(k)}[ \cup ]d^{(k)}, +\infty[. \quad (4.10)$$

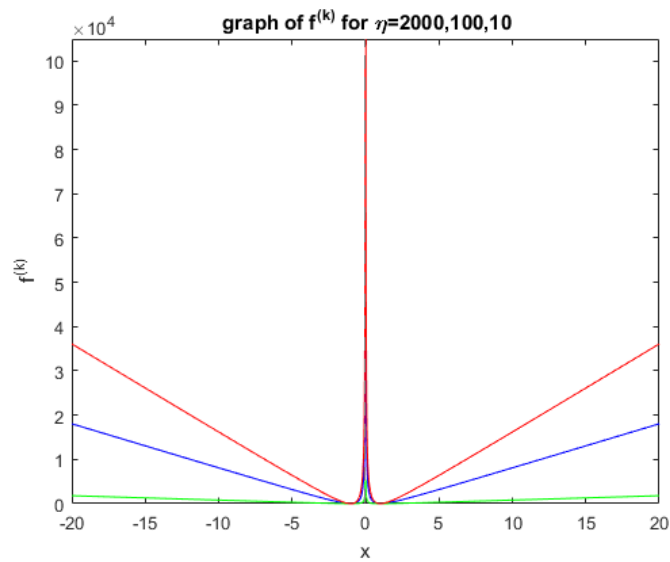
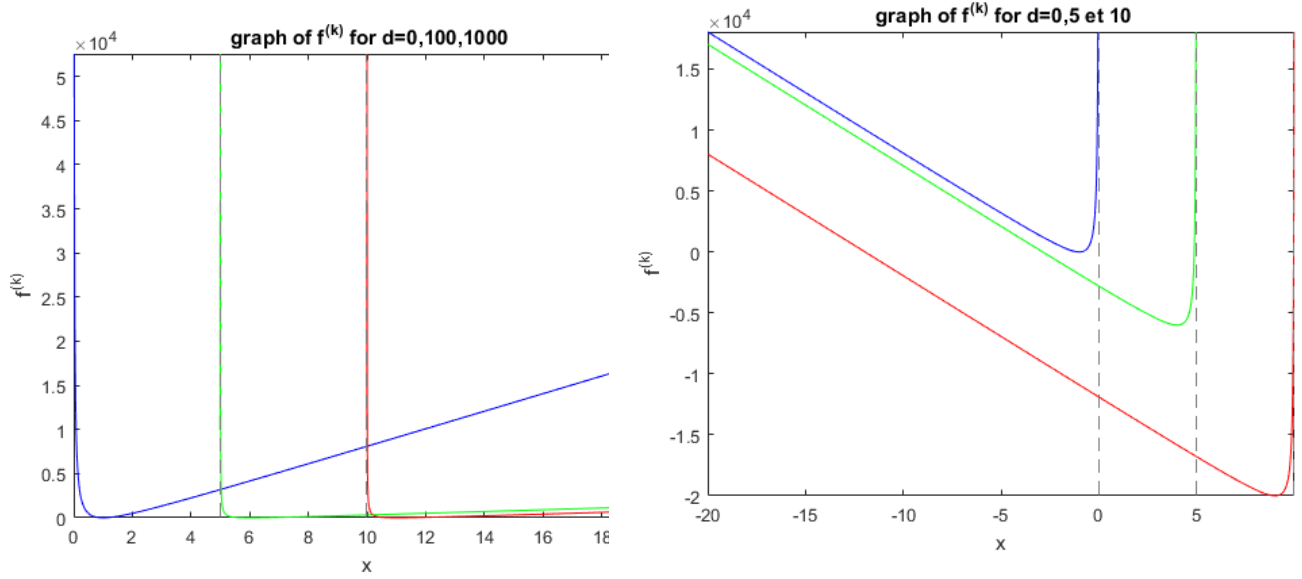


Figure 4.1 Graph of  $f^{(k)}$  for different values of  $\eta$

Figure 4.2 Graph of  $f^{(k)}$  for different values of  $d^{(k)}$ 

We now are able to state the first result:

**Theorem 4.2.1** *In above notation, let  $\Omega \subset \mathbb{R}$  be an open set of the real line, a given twice differentiable strictly convex in  $\Omega$ ,  $x_0 \in \Omega$  and  $x^{(k)}$  being respectively the initial and a current point of the sequence  $\{x^{(k)}\}_{k \geq 0}$ . Let the choice of the asymptotes be feasible. Then for each  $k \geq 0$  the approximated function  $f^{(k)}$  is a strictly convex function on  $I^{(k)}$ . In addition, the function  $f^{(k)}$  has an unique minimum at*

$$x_k^* = d^{(k)} + S^{(k)} * \sqrt{(x^{(k)} - d^{(k)})^2 \left(1 - \frac{g(x^{(k)})}{\frac{\eta^{(k)}}{2}(x^{(k)} - d^{(k)})}\right)}. \quad (4.11)$$

**Proof 9** *An important characteristic of our approximate problem obtained via th approximation function  $f^{(k)}$  is its strict convexity in  $I^{(k)}$ . But from the definition of  $I^{(k)}$  and a simple calculation of its second derivative can be written in the form:*

$$\left(f^{(k)}\right)''(x) = \left(\eta^{(k)} - \frac{2b^{(k)}}{x^{(k)} - d^{(k)}}\right) \left(\frac{x^{(k)} - d^{(k)}}{x - d^{(k)}}\right)^3. \quad (4.12)$$

Hence, to prove the convexity of  $f^{(k)}$ , we have to show that

$$\left(\eta^{(k)} - \frac{2b^{(k)}}{x^{(k)} - d^{(k)}}\right) \left(\frac{x^{(k)} - d^{(k)}}{x - d^{(k)}}\right)^3 > 0, \quad \forall x \in \mathcal{I}^{(k)}.$$

According to the definition of the set  $\mathcal{I}^{(k)}$ , it follows that  $x^{(k)} - d^{(k)}$  and  $x - d^{(k)}$  have the same sign in the interval  $\mathcal{I}^{(k)}$ , It remains to show that,

$$\left(\eta^{(k)} - \frac{2b^{(k)}}{x^{(k)} - d^{(k)}}\right) > 0. \quad (4.13)$$

If we replace  $x^{(k)} - d^{(k)}$  by  $\sigma^{(k)}$  in 4.13

$$\eta^{(k)} - 2 \frac{g(x^{(k)})}{x^{(k)} - d^{(k)}} = \eta^{(k)} - 2 \frac{g(x^{(k)})}{\sigma^{(k)}} \quad (4.14)$$

$$= \eta^{(k)} - 2 \frac{g(x^{(k)})}{\sigma^{(k)}} > 0 \quad (4.15)$$

According to 4.15 we conclude that the inequality (4.13) satisfied, and then we immediately get the strict convexity of  $f^{(k)}$  on  $\mathcal{I}^{(k)}$ . Furthermore, if  $\tilde{f}^{(k)}$  attains its minimum at  $x_*^{(k)}$ , then it is easy to see that  $x_*^{(k)}$  is solution of the equation

$$\frac{\eta^{(k)}}{2} (x^{(k)} - d^{(k)}) + \frac{(x^{(k)} - d^{(k)})^2}{(x - d^{(k)})^2} \left[ g(x^{(k)}) - \frac{\eta^{(k)}}{2} (x^{(k)} - d^{(k)}) \right] = 0 \quad (4.16)$$

From this we get the equivalent equation

$$\begin{aligned} \frac{(x^{(k)} - d^{(k)})^2}{(x - d^{(k)})^2} &= \frac{\frac{\eta^{(k)}}{2} (x^{(k)} - d^{(k)})}{\frac{\eta^{(k)}}{2} (x^{(k)} - d^{(k)}) - g(x^{(k)})} \\ &= \frac{1}{1 - \frac{2g(x^{(k)})}{\eta^{(k)}(x^{(k)} - d^{(k)})}} \\ &= \frac{1}{1 - \frac{2g(x^{(k)})}{2g(x^{(k)}) + \eta^{(k)}(x^{(k)} - d^{(k)})}} \\ &= \frac{1}{1 - \frac{1}{1 + \frac{\eta^{(k)}}{2g(x^{(k)})} (x^{(k)} - d^{(k)})}} \end{aligned}$$

As the sequence of asymptotes is feasible, then  $(x^{(k)} - d^{(k)})$  and  $g(x^{(k)})$  have the same sign, and  $\eta^{(k)} > 0 \quad \forall k > 0$   
So

$$\frac{\eta^{(k)}}{2g(x^{(k)})} (x^{(k)} - d^{(k)}) > 0$$

$$1 + \frac{\eta^{(k)}}{2g(x^{(k)})} (x^{(k)} - d^{(k)}) > 1$$

$$0 < \frac{1}{1 + \frac{\eta^{(k)}}{2g(x^{(k)})} (x^{(k)} - d^{(k)})} < 1$$

$$1 - \frac{1}{1 + \frac{\eta^{(k)}}{2g(x^{(k)})} (x^{(k)} - d^{(k)})} > 0$$

Hence if  $x_*^{(k)}$  the root of  $f^{(k)}$ , then we get that the point  $x_*^{(k)}$  satisfies the equation

$$\left(x_*^{(k)} - d^{(k)}\right)^2 = \left(x^{(k)} - d^{(k)}\right)^2 \left(1 - \frac{2b^{(k)}}{\eta^{(k)}(x^{(k)} - d^{(k)})}\right)$$

Now, by taking the square root and using the definition of feasible asymptotes we can see that the unique solution  $x_*^{(k)}$  belonging to  $I^{(k)}$  is given by

$$x_k^* = d^{(k)} + S^{(k)} \sqrt{\left(x^{(k)} - d^{(k)}\right)^2 \left(1 - \frac{b^{(k)}}{\frac{\eta^{(k)}}{2}(x^{(k)} - d^{(k)})}\right)}.$$

**Remark 4.2.2** An analysis of the previous proofs reveals that instead of assuming that the objective function is convex, it suffices to only require that we have the condition

$$\left(\eta^{(k)} - \frac{2b^{(k)}}{x^{(k)} - d^{(k)}}\right) > 0, \quad \forall k > 0. \quad (4.17)$$

**Remark 4.2.3** It is easily seen that our approximation can be seen as an extension of the approximation of Fleury, where in our method we use the second order information.

We have built a strictly convex approximation function of the objective function  $f$  at  $x^{(k)}$ , which is minimum at  $x_*^{(k)} \in I^{(k)}$ . We have now to check that the sequence  $(x^{(k)})$  defined by

$$x^{(k+1)} = x_*^{(k)}$$

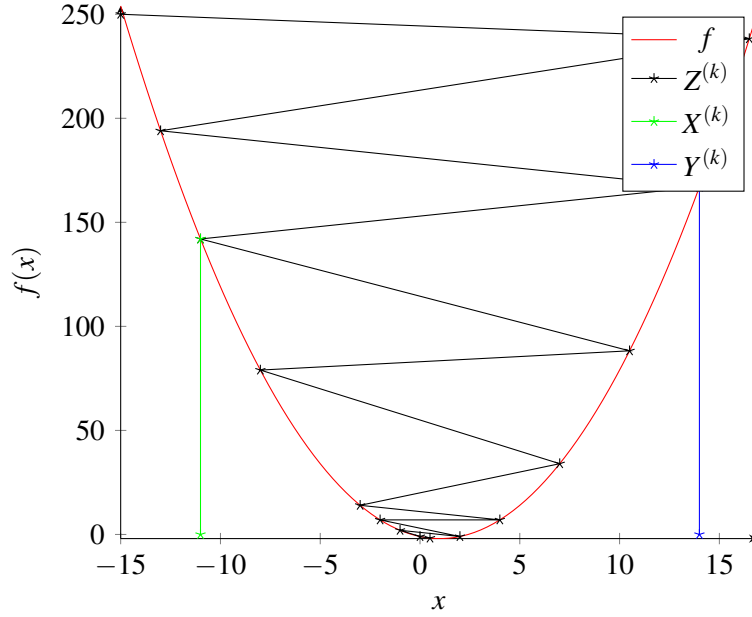
converges to the solution  $x_*$  of the initial problem. On the relationship between the new method and several of the most closely related ideas. Our approximation scheme leaves, as in the one-dimensional case, all well-known properties of convexity and separability of the MMA unchanged with the following major advantages:

- All our subproblems have an explicit solutions.
- It converges an iteration sequence that is bounded and converges to a local solution.
- It converges geometrically.

To simplify the notation, for every  $j = 1, \dots, d$ , we use  $f_{.j}$  to denote the first order partial derivative of  $f$  with respect to each variable  $x_j$ , we also use the notation  $f_{.,ij}$  for the second-order partial derivatives with respect to  $x_i$  first and then  $x_j$ ,

### 4.2.2 Convergence analysis in $\mathbb{R}^d$

This subsection aims to show that the proposed method is convergent in the sense that the distance between the both sub sequence  $\{X^{(k)}\}$  and  $\{Y^{(k)}\}$  generated by theorem 4.2.1 decrease when the iteration  $k$  increase and the distance between two successive distances converges to zero

Figure 4.3 Behavior of the sequence  $x^{(k)}$ 

**Definition 4.2.4** *Local convergence* An iterative method with iteration function  $f : \Omega \subset \mathbb{R}^k \rightarrow \mathbb{R}$  converges locally to  $x_* \in \mathbb{R}^k$  if there exists a neighborhood  $S \subset \mathbb{R}^k$  of  $x_*$  such that for every  $x^{(0)} \in S \cup \Omega$  it holds that

$$\lim_{k \rightarrow +\infty} x^{(k)} = x_*$$

where  $\{x^{(k)}\}_{k \in \mathbb{N}}$  are the iterates associated to the iterative sequence defined in theorem 4.4.3

**Theorem 4.2.5** Let consider the sequence define by Theorem 4.2.1, Assume that  $\nabla f(x_*) = 0$  for some  $x_* \in \Omega$ . Then there exists an open interval  $S = [L_{min}, U_{max}] \subset \Omega$  containing  $x_*$  such that , for any  $x_0 \in S$ , the scheme (4.11) are well defined, remain in  $S$  and converge to  $x_*$ .

**Proof 10** Let  $\mu$  be a fixed positive constant such that

$$0 < \frac{1}{\eta^{(k)}} \leq \mu \quad \forall k \geq 0 \quad (4.18)$$

and we define the sequence  $x^{(k+1)}$  in the theorem (4.2.1) by

$$x^{(k+1)} = \begin{cases} d^{(k)} + \sqrt{(x^{(k)} - d^{(k)})^2 (1 - 2 \frac{g(x^{(k)})}{\eta^{(k)}(x^{(k)} - d^{(k)})})} & \text{if } g(x^{(k)}) < 0, \\ d^{(k)} - \sqrt{(x^{(k)} - d^{(k)})^2 (1 - 2 \frac{g(x^{(k)})}{\eta^{(k)}(x^{(k)} - d^{(k)})})} & \text{if } g(x^{(k)}) > 0. \end{cases}$$

$$= \begin{cases} X^{(k+1)} = \{x^{(k+1)} / g(x^{(k+1)}) < 0\}, \\ Y^{(k+1)} = \{x^{(k+1)} / g(x^{(k+1)}) > 0\}, \end{cases}$$

If we suppose that  $g(x^{(0)}) > 0$  (we can of course suppose the other case, just the notification will be changed)

$$\begin{aligned} x^{(k+1)} &= \begin{cases} X^{(k+1)} = \{x^{2(p+1)}, & p \in \mathbb{N}\}, \\ Y^{(k+1)} = \{x^{2(p+1)+1}, & p \in \mathbb{N}\}, \end{cases} \\ &= \begin{cases} d^{(k)} + \sqrt{(X^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(X^{(k)})}{\eta^{(k)}(X^{(k)} - d^{(k)})}\right)}, \\ d^{(k)} - \sqrt{(Y^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(Y^{(k)})}{\eta^{(k)}(Y^{(k)} - d^{(k)})}\right)}. \end{cases} \end{aligned}$$

$X^{(k)}$  and  $Y^{(k)}$  are a both sub-sequence of the  $x^{(k)}$  such that  $X^{(k)} = \{x^{2p}\}_{p \in \mathbb{N}}$  and  $Y^{(k)} = \{x^{2p+1}\}_{p \in \mathbb{N}}$  and

$$\begin{aligned} g(X^{(k)}) &> 0 \quad \forall k \in \mathbb{N}, \\ g(Y^{(k)}) &< 0 \quad \forall k \in \mathbb{N}, \end{aligned} \quad (4.19)$$

$$d^{(k)} = \begin{cases} X^{(k)} + \sigma^{(k)}, \\ Y^{(k)} + \sigma^{(k)}, \end{cases} \quad (4.20)$$

$$X^{(k+1)} - Y^{(k+1)} = X^{(k)} - Y^{(k)} + \sqrt{(X^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(X^{(k)})}{\eta^{(k)}(X^{(k)} - d^{(k)})}\right)} \quad (4.21)$$

$$+ \sqrt{(Y^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(Y^{(k)})}{\eta^{(k)}(Y^{(k)} - d^{(k)})}\right)} \quad (4.22)$$

Let us note by

$$G_X^{(k)} = (X^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(X^{(k)})}{\eta^{(k)}(X^{(k)} - d^{(k)})}\right) \quad (4.23)$$

$$G_Y^{(k)} = (Y^{(k)} - d^{(k)})^2 \left(1 - 2 \frac{g(Y^{(k)})}{\eta^{(k)}(Y^{(k)} - d^{(k)})}\right) \quad (4.24)$$

So

$$X^{(k+1)} - Y^{(k+1)} = X^{(k)} - Y^{(k)} + \sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}} \quad (4.25)$$

$$\left| X^{(k+1)} - Y^{(k+1)} \right| < \left| X^{(k)} - Y^{(k)} \right| + \left| \sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}} \right| \quad (4.26)$$

$\{X^{(k)}\}$  is an increase sequence, indeed

$$X^{(k+1)} - X^{(k)} = \sigma^{(k)} + \sqrt{G_X^{(k)}} > 0 \quad \forall k \geq 0, \quad (4.27)$$



and  $\{Y^{(k)}\}$  is a decrease sequence,

$$Y^{(k+1)} - Y^{(k)} = \sigma^{(k)} - \sqrt{G_Y^{(k)}} < 0 \quad \forall k \geq 0 \quad (4.28)$$

According to (4.6), (4.15) and (4.79) we have

$$\frac{g(Y^{(k)})}{\eta^{(k)}} \sigma^{(k)} < 0 \quad (4.29)$$

then

$$Y^{(k+1)} - Y^{(k)} < 0 \quad \forall k \geq 0, \quad (4.30)$$

Let us consider the table of variation of the function  $f$  in the neighborhood of  $x_*$

$x$	$L_{min}$	$x_*$	$U_{max}$
$g(x)$		0	
$f(x)$		$f_{min}$	

Table 4.1 Variation table of  $f$

From (4.79) and table 4.1

$$X^{(k)} \in ]L_{min}, x_*[ \quad \forall k \geq 0, \quad (4.31)$$

$$Y^{(k)} \in ]x_*, U_{max}[ \quad \forall k \geq 0, \quad (4.32)$$

and then

$$Y^{(k)} > X^{(k)} \quad \forall k \geq 0, \quad (4.33)$$

From (4.27), (4.30) and (4.91) we can easily check that

$$0 < Y^{(k+1)} - X^{(k+1)} < Y^{(k)} - X^{(k)} \quad \forall k \geq 0, \quad (4.34)$$

$$Y^{(k+1)} - X^{(k+1)} = Y^{(k)} - X^{(k)} - (\sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}}) \quad (4.35)$$

$$|Y^{(k+1)} - X^{(k+1)}| = |Y^{(k)} - X^{(k)}| - (\sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}}) \quad (4.36)$$

$$|Y^{(k+1)} - X^{(k+1)}| - |Y^{(k)} - X^{(k)}| = -(\sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}}) \quad (4.37)$$

According to the last equality (4.37), we conclude

$$|Y^{(k+1)} - X^{(k+1)}| < |Y^{(k)} - X^{(k)}| \quad \forall k \geq 0 \quad (4.38)$$

If we pose  $Z^{(k)} = |Y^{(k)} - X^{(k)}|$

$$\begin{aligned} |Z^{(k+1)} - Z^{(k)}|^2 &= (\sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}})^2 \\ &= G_X^{(k)} + G_Y^{(k)} + 2\sqrt{G_X^{(k)}}\sqrt{G_Y^{(k)}} \end{aligned}$$

Therefore

$$\begin{aligned} G_X^{(k)} + G_Y^{(k)} &= (X^{(k)} - d^{(k)})^2 \left(1 - 2\frac{g(X^{(k)})}{\eta^{(k)}(X^{(k)} - d^{(k)})}\right) + (Y^{(k)} - d^{(k)})^2 \left(1 - 2\frac{g(Y^{(k)})}{\eta^{(k)}(Y^{(k)} - d^{(k)})}\right) \\ &= \left(X^{(k)} - d^{(k)} - \frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(Y^{(k)} - d^{(k)} - \frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2 - \left(\left(\frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(\frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2\right) \\ &= \left(\sigma^{(k)} - \frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(\sigma^{(k)} - \frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2 - \left(\left(\frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(\frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2\right) \\ &= \left(\sigma^{(k)} - \frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(\sigma^{(k)} - \frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2 - \left(\left(\frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + \left(\frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2\right) \\ &< (2M_1 - 1)^2 \left(\frac{g(X^{(k)})}{\eta^{(k)}}\right)^2 + (2M_2 + 1)^2 \left(\frac{g(Y^{(k)})}{\eta^{(k)}}\right)^2 \\ &< \frac{(2M - 1)^2}{(\eta^{(k)})^2} (g(X^{(k)}))^2 + \frac{(2M - 1)^2}{(\eta^{(k)})^2} (g(Y^{(k)}))^2 \\ &< \frac{(2M - 1)^2}{(\eta^{(k)})^2} \left[ (g(X^{(k)}))^2 + (g(Y^{(k)}))^2 \right] \end{aligned}$$

Now by using the first identity of Legendre we find

$$\begin{aligned} |Z^{(k+1)} - Z^{(k)}|^2 &= (\sqrt{G_X^{(k)}} + \sqrt{G_Y^{(k)}})^2 \\ &= G_X^{(k)} + G_Y^{(k)} + 2\sqrt{G_X^{(k)}}\sqrt{G_Y^{(k)}} \\ &< 2(G_X^{(k)} + G_Y^{(k)}) \end{aligned}$$

so

$$0 < \|Z^{(k+1)} - Z^{(k)}\| < \frac{2M - 1}{\eta^{(k)}} \left[ \|g(X^{(k)})\| + \|g(Y^{(k)})\| \right], \quad (4.39)$$

$$(4.40)$$

$$< (2M - 1)\mu \left[ \|g(X^{(k)})\| + \|g(Y^{(k)})\| \right], \quad (4.41)$$

or  $\{X^{(k)}\}_{k \in \mathbb{N}}$  and  $\{Y^{(k)}\}_{k \in \mathbb{N}}$  two sub-sequences of  $\{x^{(k)}\}_{k \in \mathbb{N}}$  then

$$\lim X^{(k)} = \lim Y^{(k)} = \lim x^{(k)} = \bar{x} \quad (4.42)$$

$g$  is a function continuously differentiable so

$$\lim g(X^{(k)}) = g(\lim X^{(k)}) = g(\bar{x}) \quad \text{and} \quad \lim g(Y^{(k)}) = g(\lim Y^{(k)}) = g(\bar{x}) \quad (4.43)$$

According to (4.79) we have

$$\begin{aligned} g(X^{(k)}) &< 0 \quad \forall k \geq 0 \\ g(Y^{(k)}) &> 0 \quad \forall k \geq 0 \end{aligned}$$

From (4.79) and (4.43) we conclude

$$g(\bar{x}) = 0, \quad \text{and} \quad \bar{x} = x_*, \quad (4.44)$$

and  $\bar{x}$  is a optimizer of the objective function  $f$ . Now from (4.27) and (4.30) and table 4.1 we can conclude that

$$\begin{aligned} &\{f(X^{(k)})\} \text{ is a decreasing sequence} \\ \text{and} & \\ &f(X^{(k)}) > f_{\min} \quad \forall k \geq 0 \end{aligned} \quad (4.45)$$

$$\begin{aligned} &\{f(Y^{(k)})\} \text{ is a decreasing sequence} \\ \text{and} & \\ &f(Y^{(k)}) > f_{\min} \quad \forall k \geq 0 \end{aligned} \quad (4.46)$$

### 4.3 The multivariate new approach of the MMA

To build up the approximate optimization subproblems  $P[k]$ , taking into account the approximate optimization problem, we will seek to construct a successive sequence of subproblems  $P[k] \quad k \in \mathbb{N}$ , at successive iteration  $k$ , we shall seek a suitable explicit rational approximating function  $f^{(k)}$ , strictly convex and relatively easy to implement. The solution of the subproblems  $P[k]$  will be obtained explicitly and is denoted by  $x - *^{(k)}$ . The optimum  $x - *^{(k)}$  of the subproblem  $P[k]$  will be considered as the starting point  $x^{(k+1)} := x - *^{(k)}$  for the next subsequent approximate subproblem  $P[k+1]$ . Therefore, for a given suitable initial approximation  $x^{(0)}$ , the approximate subproblem  $P[k]$ . To reduce the computational cost, the Hessian of the objective function at each iteration will be replaced by a sequence of diagonal Hessian estimates. These approximate matrix use only first order information accumulated during the previous iterations. However, in view of practical difficulties of evaluating the second-order information, a fitting algorithmic scheme is proposed in order to adjust the curvature of the approximation. The purpose of the first part of this section is to give a complete discussion on the theoretical aspects concerning the multivariate setting. We will first describe the setup and notation for our approach.

### 4.3.1 The multivariate setting

To develop our method for multivariate case, we need to replace the approximating functions of the univariate objective function by suitable strictly convex multivariate approximating functions. The practical implementation of this method is considerably more complex than in the univariate case due to the fact that, at each iteration, the approximating function in the multivariate setting generate a sequence of diagonal Hessian estimates. In this section as well as in the univariate objective approximating function presented in previews section, the function value  $f(x^{(k)})$ , the first order derivatives  $\frac{\partial f}{\partial x_j}(x^{(k)})$ , for  $j = 1, \dots, d$ , as well as the spectral parameters and the moving asymptotes at the design point  $x^{(k)}$  are used to build up effort, we suggest to approximate at each iteration the second derivative  $f''(x^{(k)})$  by some positive real value  $s^{(k)}$ . In this situation, we shall propose the following procedure for selecting moving asymptotes

$$d^{(k)} = \begin{cases} x^{(k)} + \sigma^{(k)}, & \sigma^{(k)} > 0 & \text{if } g(x^{(k)}) > 0, \\ x^{(k)} + \sigma^{(k)}, & \sigma^{(k)} < 0 & \text{if } g(x^{(k)}) < 0. \end{cases}$$

It is clear that all the previews results easily carry over to the case in the proposition, the second derivative  $f''(x^{(k)})$  is replaced by an approximate strictly positive parameters  $\eta^{(k)}$  according to the constraints. Indeed, the statements of the theorem apply with straight forward changes. In the next section for the multivariate case, we will discuss a strategy to determinate at each iteration a reasonably good numerical approximation to the second derivative. we will also establish a multivariate version of theorem. We now give a short discussion about an extension of the above approach. Our study in this section has been in a framework that at each iteration, the second derivative need to be evaluate exactly. We will focus our analysis on examining what happen when the second derivative of the objective function  $f$  may not be known or is expensive to evaluate. Thus in order to reduce the computational cost.

#### 4.4 Presentation of our method in $\mathbb{R}^\infty$

In this section a generalization of our method to  $IR^n$  is presented. The objective function being in general form  $f : IR^n$ . We introduce for  $x \in \Omega \subset IR^n$ , the following approximating functions:

$$\begin{aligned}
f^{(k)}(x) &= f(x^{(k)}) + \sum_{i=1}^n \frac{\partial f}{\partial x_i}(x^{(k)}) \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) (x_i^{(k)} - d_i^{(k)})^2 \\
&+ \sum_{i=1}^n \frac{\eta_i^{(k)}}{2} \left[ \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right] \\
&= f(x^{(k)}) + \sum_{i=1}^n g_i(x^{(k)}) \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) (x_i^{(k)} - d_i^{(k)})^2 \\
&+ \sum_{i=1}^n \frac{\eta_i^{(k)}}{2} \left[ \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right] \\
&= f(x^{(k)}) + \nabla f(x^{(k)})^T \cdot v(x, x^{(k)}, d^{(k)}) + \eta^{(k)} \cdot w(x, x^{(k)}, d^{(k)}) \\
&= f(x^{(k)}) + \nabla f(x^{(k)})^T \cdot v^{(k)}(x) + \eta^{(k)} \cdot w^{(k)}(x),
\end{aligned} \tag{4.47}$$

where  $v^{(k)} : IR^n \longrightarrow IR^n$  and  $w^{(k)} : IR^n \longrightarrow IR^n$  two functions define by

$$\begin{aligned}
v^{(k)} : IR^n &\longrightarrow IR^n \\
x &\longrightarrow (v_1^{(k)}(x_1), \dots, v_n^{(k)}(x_n)),
\end{aligned} \tag{4.48}$$

$$\begin{aligned}
w^{(k)} : IR^n &\longrightarrow IR^n \\
x &\longrightarrow (w_1^{(k)}(x_1), \dots, w_n^{(k)}(x_n)),
\end{aligned} \tag{4.49}$$

where for  $i = 1, \dots, n$

$$\begin{aligned}
v_i^{(k)}(x_i) &= \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) (x_i^{(k)} - d_i^{(k)})^2 \\
w_i^{(k)}(x_i) &= \left[ \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right].
\end{aligned} \tag{4.50}$$

The approximate functions  $f^{(k)}$  are define on

$$\mathcal{S}^{(k)} = \mathcal{S}_1^{(k)} \times \dots \times \mathcal{S}_n^{(k)} \tag{4.51}$$

$$\mathcal{S}_i^{(k)} = ]-\infty, d_i^{(k)}[ \cup ]d_i^{(k)}, +\infty[, \quad i = 1, \dots, n \tag{4.52}$$

We can check easily that

$$\begin{aligned} v^{(k)}(x^{(k)}) &= w^{(k)}(x^{(k)}) = \mathbf{0}_{\mathbb{R}^n} \\ Dv^{(k)}(x^{(k)}) &= Dw^{(k)}(x^{(k)}) = \mathbf{0}_{\mathbb{R}^n}, \end{aligned} \quad (4.53)$$

with  $Dv^{(k)} = (Dv_1^{(k)}, \dots, Dv_n^{(k)})$  and  $Dw^{(k)} = (Dw_1^{(k)}, \dots, Dw_n^{(k)})$ .

From (4.53), we can see that the functions define in (4.47) are an approximations of first order and we have

$$\begin{aligned} f^{(k)}(x^{(k)}) &= f(x^{(k)}) \\ \nabla f^{(k)}(x^{(k)}) &= \nabla f(x^{(k)}) \end{aligned} \quad (4.54)$$

The first and second derivatives of the convex approximations can be given analytically by

$$\nabla f^{(k)}(x) = \begin{pmatrix} g_1(x^{(k)}) \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} + \frac{\eta_1^{(k)}}{2} (x_1^{(k)} - d_1^{(k)}) \left( 1 - \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} \right) \\ \vdots \\ g_i(x^{(k)}) \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} + \frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) \left( 1 - \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \right) \\ \vdots \\ g_n(x^{(k)}) \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} + \frac{\eta_n^{(k)}}{2} (x_n^{(k)} - d_n^{(k)}) \left( 1 - \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} \right) \end{pmatrix} \quad (4.55)$$

$$\frac{\partial f^{(k)}}{\partial x_i}(x) := g_i(x^{(k)}) \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} + \frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) \left( 1 - \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \right) \quad (4.56)$$

$$\frac{\partial^2 f^{(k)}(x)}{\partial x_i \partial x_j} = 0, \quad \forall i \neq j \quad (4.57)$$

$$\frac{\partial^2 f^{(k)}}{\partial x_i^2}(x) := -g_i(x^{(k)}) \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^3} + \eta_i^{(k)} \frac{(x_i^{(k)} - d_i^{(k)})^3}{(x_i - d_i^{(k)})^3}. \quad (4.58)$$

We suppose

$$\sigma_i^{(k)} > 2 \frac{|g_i(x^{(k)})|}{\eta_i^{(k)}} > 0 \quad (4.59)$$

$$\nabla^2 f^{(k)}(x) = \begin{pmatrix} \frac{\partial^2 f^{(k)}}{\partial x_1^2}(x) & 0 & \cdots & 0 \\ 0 & \frac{\partial^2 f^{(k)}}{\partial x_2^2}(x) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \frac{\partial^2 f^{(k)}}{\partial x_n^2}(x) \end{pmatrix}, \quad (4.60)$$

We can rewrite the Hessian matrix of  $f^{(k)}$  as

$$\nabla^2 f^{(k)}(x) = A^{(k)} \cdot B^{(k)}(x) \quad (4.61)$$

where  $(\cdot)$  is the Hadamard product of the diagonals matrix  $A^{(k)}$  and  $B^{(k)}$  defined by

$$A^{(k)} = \begin{pmatrix} \left( \eta_1^{(k)} - 2 \frac{g_1(x^{(k)})}{x_1^{(k)} - d_1^{(k)}} \right) & 0 & \cdots & 0 \\ 0 & \left( \eta_2^{(k)} - 2 \frac{g_2(x^{(k)})}{x_2^{(k)} - d_2^{(k)}} \right) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \left( \eta_n^{(k)} - 2 \frac{g_n(x^{(k)})}{x_n^{(k)} - d_n^{(k)}} \right) \end{pmatrix} \quad (4.62)$$

$$B^{(k)}(x) = \begin{pmatrix} \left( \frac{x_1^{(k)} - d_1^{(k)}}{x_1 - d_1^{(k)}} \right)^3 & 0 & \cdots & 0 \\ 0 & \left( \frac{x_2^{(k)} - d_2^{(k)}}{x_2 - d_2^{(k)}} \right)^3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \left( \frac{x_n^{(k)} - d_n^{(k)}}{x_n - d_n^{(k)}} \right)^3 \end{pmatrix} \quad (4.63)$$

**Definition 4.4.1** A sequence of asymptotes  $\{d^{(k)}\} = \{d_1^{(k)}, \dots, d_n^{(k)}\}$  defined by

$$d_i^{(k)} = x_i^{(k)} + \sigma_i^{(k)}, \quad i = 1, \dots, n. \quad (4.64)$$

is called feasible if for all  $k \geq 0$ ,

$$\begin{aligned} \sigma_i^{(k)} &> 2 \frac{g_i(x^{(k)})}{\eta_i^{(k)}} > 0, \quad \text{if } g_i(x^{(k)}) > 0, \quad i = 1, \dots, n \\ \sigma_i^{(k)} &< 2 \frac{g_i(x^{(k)})}{\eta_i^{(k)}} < 0, \quad \text{if } g_i(x^{(k)}) < 0, \quad i = 1, \dots, n. \end{aligned} \quad (4.65)$$

**Proposition 4.4.2** Let  $\{d^{(k)}\} = \{d_1^{(k)}, \dots, d_n^{(k)}\}$  be a sequence of asymptotes,  $M$  a constant such that  $M > 1$  and let the assumptions 4.65 be valid. Then  $\{d^{(k)}\}$  is feasible if and only if

$$\sigma_i^{(k)} = 2M\alpha_k \frac{g_i(x^{(k)})}{\eta_i^{(k)}}, \quad i = 1, \dots, n. \quad (4.66)$$

where the positive scalar  $\alpha_k$  is called the step length, this parameter is used to stabilize the convergence.

**Theorem 4.4.3** With the above notation, let  $\Omega \subset \mathbb{R}^n$  be an open subset, a given continuously differentiable function  $f$  in  $\Omega$ ,  $x^{(0)} \in \Omega$  and  $x^{(k)}$  being respectively the initial and a current point of the sequence  $\{x^{(k)}\}_{k \geq 0}$ . Then, for each  $k > 0$  the approximating function defined by (4.47) is a strictly convex function on  $\mathcal{S}_k$ . In addition, the function  $f^{(k)}$  has a unique minimum at

$$x^{(k+1)} \leftarrow x_*^{(k)} = d^{(k)} + S^{(k)} G^{(k)} \quad (4.67)$$

where  $G^{(k)} = \text{diag}(G_1^{(k)}, \dots, G_n^{(k)})$  and  $S^{(k)} = (S_1^{(k)}, \dots, S_n^{(k)})^T$

$$G_i^{(k)} = \sqrt{(x_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(x^{(k)})}{\eta_i^{(k)} (x_i^{(k)} - d_i^{(k)})}\right)}. \quad (4.68)$$

and

$$S_i^{(k)} = \begin{cases} -1 & \text{if } g_i(x^{(k)}) < 0, \\ 1 & \text{if } g_i(x^{(k)}) > 0. \end{cases} \quad (4.69)$$

**Proof 11** . The main ingredient here is a suitable application of the condition (4.59) imposed on the coefficient  $\sigma^{(k)}$ . We first start by showing that the approximating function  $f^{(k)}$  is well defined and strictly convex in  $\mathcal{S}_k$ . To this end we prove that  $\nabla^2 f^{(k)}$  is positive semidefinite in  $\mathcal{S}_k$ . Indeed, a simple calculation reveals that

$$\nabla^2 f^{(k)}(x) = \begin{pmatrix} \Gamma_1^{(k)} \left( \frac{x_1^{(k)} - d_1^{(k)}}{x_1 - d_1^{(k)}} \right)^3 & 0 & \dots & 0 \\ 0 & \Gamma_2^{(k)} \left( \frac{x_2^{(k)} - d_2^{(k)}}{x_2 - d_2^{(k)}} \right)^3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \Gamma_n^{(k)} \left( \frac{x_n^{(k)} - d_n^{(k)}}{x_n - d_n^{(k)}} \right)^3 \end{pmatrix}, \quad (4.70)$$

where

$$\Gamma_i^{(k)} = \eta_i^{(k)} - 2 \frac{g_i(x^{(k)})}{x_i^{(k)} - d_i^{(k)}} \quad i = 1, \dots, n. \quad (4.71)$$

In view of (4.70), it remains to show that the Hessian matrix on the right-hand side of (4.70) is positive semidefinite for all  $x \in \mathcal{S}_k$ . Since  $\gamma^{(k)}$  is nonnegative (as can be seen in (4.59)), and the two terms  $x_i^{(k)} - d_i^{(k)}$



and  $x_i - d_i^{(k)}$  have the same sign in the interval  $\mathcal{I}_j^k$ , and  $\forall x \in \mathcal{I}_j^k, x^T \cdot \nabla^2 f^{(k)}(x) \cdot x \geq 0$  then  $f^{(k)}$  is a convex function on  $\mathcal{I}_k$ . Furthermore, the function  $f^{(k)}$  being well definite and continuous in  $\mathcal{I}_k$ , this implies the existence of a minimum, which by convexity is the unique critical point  $x_*^{(k)}$ . Now, looking for  $\nabla f^{(k)}(x) = 0_{\mathbb{R}^n}$  we conclude that the optimum  $x_*^{(k)}$  is one solution of the system

$$\begin{pmatrix} g_1(x^{(k)}) \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} + \frac{\eta_1^{(k)}}{2} (x_1^{(k)} - d_1^{(k)}) \left( 1 - \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} \right) \\ \vdots \\ g_i(x^{(k)}) \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} + \frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) \left( 1 - \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \right) \\ \vdots \\ g_n(x^{(k)}) \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} + \frac{\eta_n^{(k)}}{2} (x_n^{(k)} - d_n^{(k)}) \left( 1 - \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} \right) \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ \vdots \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} \quad (4.72)$$

$$g_i(x^{(k)}) \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} + \frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) \left( 1 - \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \right) = 0, \quad i = 1, \dots, n \quad (4.73)$$

which, after trivial calculations, implies

$$\begin{pmatrix} \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} \\ \vdots \\ \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \\ \vdots \\ \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} \end{pmatrix} = \begin{pmatrix} \frac{\frac{\eta_1^{(k)}}{2} (x_1^{(k)} - d_1^{(k)})}{\frac{\eta_1^{(k)}}{2} (x_1^{(k)} - d_1^{(k)}) - g_1^{(k)}} \\ \vdots \\ \frac{\frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)})}{\frac{\eta_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) - g_i^{(k)}} \\ \vdots \\ \frac{\frac{\eta_n^{(k)}}{2} (x_n^{(k)} - d_n^{(k)})}{\frac{\eta_n^{(k)}}{2} (x_n^{(k)} - d_n^{(k)}) - g_n^{(k)}} \end{pmatrix} \quad (4.74)$$

We can rewrite the system (4.76) as

$$\begin{pmatrix} \frac{(x_1^{(k)} - d_1^{(k)})^2}{(x_1 - d_1^{(k)})^2} \\ \vdots \\ \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \\ \vdots \\ \frac{(x_n^{(k)} - d_n^{(k)})^2}{(x_n - d_n^{(k)})^2} \end{pmatrix} = \begin{pmatrix} \frac{1}{1 - \frac{1}{1 + \frac{\eta_1^{(k)}}{2g_1(x^{(k)})}(x_1^{(k)} - d_1^{(k)})}} \\ \vdots \\ \frac{1}{1 - \frac{1}{1 + \frac{\eta_i^{(k)}}{2g_i(x^{(k)})}(x_i^{(k)} - d_i^{(k)})}} \\ \vdots \\ \frac{1}{1 - \frac{1}{1 + \frac{\eta_n^{(k)}}{2g_n(x^{(k)})}(x_n^{(k)} - d_n^{(k)})}} \end{pmatrix} \quad (4.75)$$

as in theorem 4.2.1 we can prove that

$$1 - \frac{1}{1 + \frac{\eta_i^{(k)}}{2g_i(x^{(k)})}(x_i^{(k)} - d_i^{(k)})} > 0 \quad i = 1, \dots, n$$

Hence if  $x_*^{(k)} = (x_{1*}^{(k)}, \dots, x_{n*}^{(k)})$  the root of  $f^{(k)}$ , then we get that the point  $x_*^{(k)}$  satisfies the system

$$\begin{pmatrix} (x_{1*}^{(k)} - d_1^{(k)})^2 \\ \vdots \\ (x_{i*}^{(k)} - d_i^{(k)})^2 \\ \vdots \\ (x_{n*}^{(k)} - d_n^{(k)})^2 \end{pmatrix} = \begin{pmatrix} (x_1^{(k)} - d_1^{(k)})^2 \left(1 - 2 \frac{g_1^{(k)}}{\eta_1^{(k)}(x_1^{(k)} - d_1^{(k)})}\right) \\ \vdots \\ (x_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i^{(k)}}{\eta_i^{(k)}(x_i^{(k)} - d_i^{(k)})}\right) \\ \vdots \\ (x_n^{(k)} - d_n^{(k)})^2 \left(1 - 2 \frac{g_n^{(k)}}{\eta_n^{(k)}(x_n^{(k)} - d_n^{(k)})}\right) \end{pmatrix} \quad (4.76)$$

and then

$$\begin{pmatrix} x_{1*}^{(k)} \\ \vdots \\ x_{i*}^{(k)} \\ \vdots \\ x_{n*}^{(k)} \end{pmatrix} = \begin{pmatrix} d_1^{(k)} \\ \vdots \\ d_i^{(k)} \\ \vdots \\ d_n^{(k)} \end{pmatrix} + \begin{pmatrix} S_1^{(k)} \sqrt{(x_1^{(k)} - d_1^{(k)})^2 \left(1 - 2 \frac{g_1^{(k)}}{\eta_1^{(k)}(x_1^{(k)} - d_1^{(k)})}\right)} \\ \vdots \\ S_i^{(k)} \sqrt{(x_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i^{(k)}}{\eta_i^{(k)}(x_i^{(k)} - d_i^{(k)})}\right)} \\ \vdots \\ S_n^{(k)} \sqrt{(x_n^{(k)} - d_n^{(k)})^2 \left(1 - 2 \frac{g_n^{(k)}}{\eta_n^{(k)}(x_n^{(k)} - d_n^{(k)})}\right)} \end{pmatrix} \quad (4.77)$$

Therefore, after some simplification, the system (4.77) becomes

$$x_*^{(k)} = d^{(k)} + S^{(k)}G^{(k)},$$

where  $G^{(k)}$  and  $S^{(k)}$  are defined respectively by (4.68) and (4.69).

By condition (4.59) imposed on the parameter  $\sigma^{(k)}$ , it can be deduced from the system (4.77) that the solvability of our subproblem can always be guaranteed. Indeed, under this condition, we will be sure that what is inside square root on the left of the system (4.77) is always nonnegative. This completes the proof of the theorem.

#### 4.4.1 Multivariate convergence result

Thanks to the separability and convexity of the approximations proposed in the multidimensional case, we can extend in a consistent way the convergence results found in the univariate case

**Theorem 4.4.4** *Let consider the sequence  $x^{(k)}$  defined by Theorem 4.4.3, Assume that  $\nabla f(x_*) = 0$  for some  $x_* \subset \Omega$ . Then there exists an open ball  $S = S_1 \times \dots \times S_n \subset \Omega$  containing  $x_*$ , where  $S_i = [l_i, u_i]$   $i = 1, \dots, n$  such that, for any  $x_0 \in S$ , the scheme (4.67) are well defined, remain in  $S$  and converge to  $x_*$ .*

**Proof 12** *Let  $\mu$  be a fixed positive constant such that*

$$0 < \frac{1}{\eta_i^{(k)}} \leq \mu, \quad i = 1, \dots, n. \quad \forall k \geq 0 \quad (4.78)$$

we can rewrite the sequence  $x^{(k+1)} = (x_1^{(k+1)}, \dots, x_n^{(k+1)})$  defined in the theorem 4.4.3 in the form

$$\begin{aligned} x_i^{(k+1)} &= \begin{cases} d_i^{(k)} + \sqrt{(x_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(x^{(k)})}{\eta_i^{(k)}(x_i^{(k)} - d_i^{(k)})}\right)} & \text{if } g_i(x^{(k)}) < 0, \\ d_i^{(k)} - \sqrt{(x_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(x_i^{(k)})}{\eta_i^{(k)}(x_i^{(k)} - d_i^{(k)})}\right)} & \text{if } g_i(x^{(k)}) > 0. \end{cases} & i = 1, \dots, n \\ &= \begin{cases} X_i^{(k+1)} = \{x_i^{(k+1)} / g_i(x^{(k+1)}) < 0\}, \\ Y_i^{(k+1)} = \{x_i^{(k+1)} / g_i(x^{(k+1)}) > 0\}, \end{cases} & i = 1, \dots, n \end{aligned}$$

Now for each dimension  $i = 1, \dots, n$ , we suppose  $g_i(x^{(0)}) > 0$

$$\begin{aligned} x_i^{(k+1)} &= \begin{cases} X_i^{(k+1)} = \{x_i^{2(p+1)}, \quad p \in \mathbb{N}\}, \\ Y_i^{(k+1)} = \{x_i^{2(p+1)+1}, \quad p \in \mathbb{N}\}, \end{cases} & i = 1, \dots, n \\ &= \begin{cases} d_i^{(k)} + \sqrt{(X_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(X^{(k)})}{\eta_i^{(k)}(X_i^{(k)} - d_i^{(k)})}\right)}, \\ d_i^{(k)} - \sqrt{(Y_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(Y^{(k)})}{\eta_i^{(k)}(Y_i^{(k)} - d_i^{(k)})}\right)}. \end{cases} & i = 1, \dots, n \end{aligned}$$

$X^{(k)} = (X_1^{(k)}, \dots, X_n^{(k)})$  and  $Y^{(k)} = (Y_1^{(k)}, \dots, Y_n^{(k)})$  are a both sub-sequence of the  $x^{(k)}$  such that  $X^{(k)} = \{x^{2p}\}_{p \in \mathbb{N}}$  and  $Y^{(k)} = \{x^{2p+1}\}_{p \in \mathbb{N}}$   
and

$$\begin{aligned} g_i(X^{(k)}) &> 0 \quad \forall k \in \mathbb{N}, \\ g_i(Y^{(k)}) &< 0 \quad \forall k \in \mathbb{N}, \end{aligned} \quad i = 1, \dots, n \quad (4.79)$$

$$d_i^{(k)} = \begin{cases} X_i^{(k)} + \sigma_i^{(k)}, \\ Y_i^{(k)} + \sigma_i^{(k)}, \end{cases} \quad i = 1, \dots, n \quad (4.80)$$

$$X_i^{(k+1)} - Y_i^{(k+1)} = X_i^{(k)} - Y_i^{(k)} + \sqrt{(X_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(X^{(k)})}{\eta_i^{(k)}(X_i^{(k)} - d_i^{(k)})}\right)} \quad (4.81)$$

$$+ \sqrt{(Y_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(Y^{(k)})}{\eta_i^{(k)}(Y_i^{(k)} - d_i^{(k)})}\right)} \quad (4.82)$$

Let us note by

$$G_{X_i}^{(k)} = (X_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(X^{(k)})}{\eta_i^{(k)}(X_i^{(k)} - d_i^{(k)})}\right) \quad (4.83)$$

$$G_{Y_i}^{(k)} = (Y_i^{(k)} - d_i^{(k)})^2 \left(1 - 2 \frac{g_i(Y^{(k)})}{\eta_i^{(k)}(Y_i^{(k)} - d_i^{(k)})}\right) \quad (4.84)$$

So

$$X_i^{(k+1)} - Y_i^{(k+1)} = X_i^{(k)} - Y_i^{(k)} + \sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}} \quad (4.85)$$

$$\left| X_i^{(k+1)} - Y_i^{(k+1)} \right| < \left| X_i^{(k)} - Y_i^{(k)} \right| + \left| \sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}} \right| \quad (4.86)$$

$\{X_i^{(k)}\}$  is an increase sequence, indeed

$$X_i^{(k+1)} - X_i^{(k)} = \sigma_i^{(k)} + \sqrt{G_{X_i}^{(k)}} > 0 \quad \forall k \geq 0, \quad (4.87)$$

and  $\{Y_i^{(k)}\}$  is a decrease sequence,

$$Y_i^{(k+1)} - Y_i^{(k)} = \sigma_i^{(k)} - \sqrt{G_{Y_i}^{(k)}} < 0 \quad \forall k \geq 0 \quad (4.88)$$

For  $a \in \Omega$  and  $1 \leq i \leq n$  fixed, we consider the ral function  $f_i$  defined by

$$f_i(x_i) = f(a_1, a_2, \dots, a_{i-1}, x_i, a_{i+1}, \dots, a_n) \quad i = 1, \dots, n$$

its definition domain being the open subset of  $\mathbb{R}$  given by

$$D(f_i) = \left\{ s \in \mathbb{R} : (a_1, a_2, \dots, a_{i-1}, s, a_{i+1}, \dots, a_n)^T \in \Omega \right\}$$

If  $f_i$  is derivable on  $a_i$ , we say that the  $i$ -th partial derivative of exist on  $a$  and is equal to  $f'_i(a_i)$ . it's noted

$$\frac{\partial f}{\partial x_i}(\mathbf{a}) \text{ ou } D_i f(\mathbf{a})$$

as

$$f'_i(a_j) = \lim_{t \rightarrow 0} \frac{f_i(a_j+t) - f_i(a_j)}{t}$$

we can see that

$$\frac{\partial f}{\partial x_j}(\mathbf{a}) = \lim_{t \rightarrow 0} \frac{f(\mathbf{a}+t\mathbf{e}_j) - f(\mathbf{a})}{t}.$$

If all the partial derivatives  $\frac{\partial f}{\partial x_1}(\mathbf{a}), \dots, \frac{\partial f}{\partial x_n}(\mathbf{a})$  exists on  $a \in \Omega$ , then the function  $f$  is differentiable on  $a$ . Now, let us  $\mathbf{v} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$ . The straight line passing through  $\mathbf{a}$  and admitting the director vector  $v$  is parametered as follows

$$\{\mathbf{a} + t\mathbf{v} : t \in \mathbb{R}\}$$

Let us the real function  $f_{\mathbf{v}}(t) = f(\mathbf{a} + t\mathbf{v})$ , its definition domain being

$$D(f_{\mathbf{v}}) = \{t \in \mathbb{R} : \mathbf{a} + t\mathbf{v} \in \Omega\}.$$

it is an open subset of  $\mathbb{R}$  that contains 0.

If  $f_{\mathbf{v}}$  is differentiable in  $t = 0$ , we say that  $f$  is differentiable in  $\mathbf{a}$  following the vector  $\mathbf{v}$ . Furthermore  $f'_{\mathbf{v}}(0)$  is called the derivative of  $\mathbf{f}$  in  $\mathbf{a}$  following  $\mathbf{v}$  or also the directional derivative of  $\mathbf{f}$  in  $\mathbf{a}$  following  $\mathbf{v}$ , and we note

$$Df(\mathbf{a}, \mathbf{v}) \text{ ou } \frac{\partial f}{\partial \mathbf{v}}(\mathbf{a})$$

So

$$Df(\mathbf{a}, \mathbf{v}) = \frac{\partial f}{\partial \mathbf{v}}(\mathbf{a}) = \lim_{t \rightarrow 0} \frac{f_{\mathbf{v}}(t) - f_{\mathbf{v}}(0)}{t} = \lim_{t \rightarrow 0} \frac{f(\mathbf{a} + t\mathbf{v}) - f(\mathbf{a})}{t}$$

If  $\mathbf{v} = \mathbf{e}_i$  and  $f$  is differentiable in  $\mathbf{a}$  following the vector  $\mathbf{e}_i$ , we can notice that  $Df(\mathbf{a}, \mathbf{e}_j) = \frac{\partial f}{\partial x_j}(\mathbf{a})$ . Thanks to the directional derivative's definition, we can define the table of variation of each real function  $f_i$  in the neighborhood of  $x_{i*}$ , for that we take  $a_j = x_{j*}$ , for  $j = 1, \dots, n$  when  $j \neq i$  and  $x_i \in [l_i, u_i]$

$x_i$	$l_i$	$x_{i*}$	$u_i$
$f'_i(x_i)$		-	+
$f_i(x_i)$		$f_{min}$	

Table 4.2 Variation table of  $f_i$   $i = 1, \dots, n$

From (4.79) and table 4.2

$$X_i^{(k)} \in ]l_i, x_{i^*}[ \quad \forall k \geq 0, \quad (4.89)$$

$$Y_i^{(k)} \in ]x_{i^*}, u_i[ \quad \forall k \geq 0, \quad (4.90)$$

and then

$$Y_i^{(k)} > X_i^{(k)} \quad \forall k \geq 0, \quad (4.91)$$

From (4.87), (4.88) and (4.91) we can easily check that

$$0 < Y_i^{(k+1)} - X_i^{(k+1)} < Y_i^{(k)} - X_i^{(k)} \quad \forall k \geq 0, \quad (4.92)$$

$$Y_i^{(k+1)} - X_i^{(k+1)} = Y_i^{(k)} - X_i^{(k)} - (\sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}}) \quad (4.93)$$

$$\left| Y_i^{(k+1)} - X_i^{(k+1)} \right| = \left| Y_i^{(k)} - X_i^{(k)} \right| - (\sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}}) \quad (4.94)$$

$$\left| Y_i^{(k+1)} - X_i^{(k+1)} \right| - \left| Y_i^{(k)} - X_i^{(k)} \right| = -(\sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}}) \quad (4.95)$$

According to the last equality (4.95), we conclude

$$\left| Y_i^{(k+1)} - X_i^{(k+1)} \right| < \left| Y_i^{(k)} - X_i^{(k)} \right| \quad \forall k \geq 0 \quad (4.96)$$

If we pose  $Z_i^{(k)} = \left| Y_i^{(k)} - X_i^{(k)} \right|$

$$\begin{aligned} \left| Z_i^{(k+1)} - Z_i^{(k)} \right|^2 &= (\sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}})^2 \\ &= G_{X_i}^{(k)} + G_{Y_i}^{(k)} + 2\sqrt{G_{X_i}^{(k)}}\sqrt{G_{Y_i}^{(k)}} \end{aligned}$$

Therefore

$$\begin{aligned}
G_{X_i}^{(k)} + G_{Y_i}^{(k)} &= (X_i^{(k)} - d_i^{(k)})^2 \left( 1 - 2 \frac{g_i(X^{(k)})}{\eta_i^{(k)}(X^{(k)} - d_i^{(k)})} \right) + (Y_i^{(k)} - d_i^{(k)})^2 \left( 1 - 2 \frac{g_i(Y^{(k)})}{\eta_i^{(k)}(Y_i^{(k)} - d_i^{(k)})} \right) \\
&= \left( X_i^{(k)} - d_i^{(k)} - \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( Y_i^{(k)} - d_i^{(k)} - \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 - \left( \left( \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 \right) \\
&= \left( \sigma_i^{(k)} - \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( \sigma_i^{(k)} - \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 - \left( \left( \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 \right) \\
&= \left( \sigma_i^{(k)} - \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( \sigma_i^{(k)} - \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 - \left( \left( \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + \left( \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 \right) \\
&< (2M_1 - 1)^2 \left( \frac{g_i(X^{(k)})}{\eta_i^{(k)}} \right)^2 + (2M_2 + 1)^2 \left( \frac{g_i(Y^{(k)})}{\eta_i^{(k)}} \right)^2 \\
&< \frac{(2M - 1)^2}{(\eta_i^{(k)})^2} (g_i(X^{(k)}))^2 + \frac{(2M - 1)^2}{(\eta_i^{(k)})^2} (g_i(Y^{(k)}))^2 \\
&< \frac{(2M - 1)^2}{(\eta_i^{(k)})^2} \left[ (g_i(X^{(k)}))^2 + (g_i(Y^{(k)}))^2 \right]
\end{aligned}$$

Now by using the first identity of Legendre we find

$$\begin{aligned}
|Z_i^{(k+1)} - Z_i^{(k)}|^2 &= (\sqrt{G_{X_i}^{(k)}} + \sqrt{G_{Y_i}^{(k)}})^2 \\
&= G_{X_i}^{(k)} + G_{Y_i}^{(k)} + 2\sqrt{G_{X_i}^{(k)}}\sqrt{G_{Y_i}^{(k)}} \\
&< 2(G_{X_i}^{(k)} + G_{Y_i}^{(k)})
\end{aligned}$$

so

$$0 < \|Z_i^{(k+1)} - Z_i^{(k)}\| < \frac{2M - 1}{\eta_i^{(k)}} \left[ \|g_i(X^{(k)})\| + \|g_i(Y^{(k)})\| \right], \quad (4.97)$$

$$(4.98)$$

$$< (2M - 1)\mu \left[ \|g_i(X^{(k)})\| + \|g_i(Y^{(k)})\| \right], \quad (4.99)$$

#### 4.4.2 Rules of updating the spectral parameters $\eta^{(k)}$ , $d^{(k)}$ and $\sigma^{(k)}$

- Parameter  $\eta^{(k)}$

The motivation for using the spectral parameter instead of the second order information in our approximations is to reduce the number of computation and evaluation at each iteration  $k$  and the oscillation of the process when  $f''(x^{(k)}) < 0$ , and also to keep the precision of the approximate solution.

From the Mean Value Theorem of the Integral Calculus we know that, given a continuously differentiable, then:

$$\nabla f(y) = \nabla f(x) + \int_0^1 \nabla^2 f(x + \alpha(y-x)) d\alpha (y-x) \quad (4.100)$$

By setting  $s = y - x$ , the scalar

$$\eta = \frac{s^T t}{s^T s} \quad (4.101)$$

where  $t = \nabla f(y) - \nabla f(x)$ , or again by the Mean Value Theorem,

$$t = \left( \int_0^1 \nabla^2 f(x + \alpha s) d\alpha \right) s,$$

defines a Rayleigh quotient with respect to the average Hessian matrix  $\left( \int_0^1 \nabla^2 f(x + \alpha s) d\alpha \right)$ . Such quotient has its value between the smallest and the largest eigenvalue of the average Hessian matrix, what motivates the terminology spectral parameter for (4.101). Thus, if we require that the Hessian of the objective function  $f$  approximated by scalar matrices, we might say that  $\eta I$  is the matrix of such type that best approximations the average Hessian. The idea is to use the spectral parameter 4.101 in the following way:

$$\nabla^2 f(x) \approx \eta I \quad \Rightarrow \quad \frac{\partial^2 f(x)}{\partial x_j^2} \approx \eta$$

The idea behind the updating of  $\rho_i$  is to increase or remain the same value of this parameter at each inner iteration, but never reduce it. Therefore, it is important that they decrease whenever an outer iteration starts to avoid slow convergence and generation of a small steps in the process. In terms of the parameters  $\sigma_j^{(k)}$ , the updating depends on the functions  $v_i^{(k)}$  and  $w_i^{(k)}$ . For this new sequential convex programming method, the Hessian matrix  $\nabla_{xx}^2 w_i^{(k)}(x)$  is diagonal with  $\frac{\partial^2 w_i^{(k)}}{\partial x_j^2}(x) \geq \frac{1}{(\sigma_j^{(k)})^2}$  for all  $j$  and  $x$ . This means that the curvature of the function  $w_i^{(k)}$  towards  $x_j$  increases as  $\sigma_j^{(k)}$  decreases. Thus, depending on the pattern of the variables in the previous iterations, they should be stabilized or released, according to the following rule. if  $k = 1$  and  $k = 2$ :

$$\sigma_j^{(k)} = \frac{x_j^{\max} - x_j^{\min}}{2}, \quad (4.102)$$

and for  $k \geq 3$ :

$$\sigma_j^{(k)} = \gamma_j^{(k)} \sigma_j^{(k-1)}, \quad (4.103)$$

where

$$\gamma_j^{(k)} = \begin{cases} 0.7 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) < 0 \\ 1.2 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) > 0 \\ 1 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) = 0 \end{cases} \quad (4.104)$$

- Parameter  $d^{(k)}$



Each iteration of the present method computes a step length  $\alpha^{(k)}$  and then decides how far to move the asymptotes  $d^{(k)}$  in terms of the step's value. The moving asymptotes expression is given by

$$d_i^{(k)} = x_i^{(k)} + 2M\alpha_k \frac{g_i(x^{(k)})}{\eta_i^{(k)}}, \quad (4.105)$$

where the positive scalar  $\alpha^{(k)}$  is called the step length. The success of the convergence process depends on effective choices of both the step length  $\alpha^{(k)}$  and then the asymptotes  $d^{(k)}$ .

Most moving asymptotes approaches require  $d^{(k)}$  to be defined in terms of two lower and upper values  $L^{(k)}$  and  $U^{(k)}$  as

$$d_i^{(k)} = \begin{cases} L_i^{(k)} & \text{if } f_{,i}(x^{(k)}) < 0 \\ U_i^{(k)} & \text{if } f_{,i}(x^{(k)}) > 0 \end{cases} \quad (4.106)$$

$$d^{(k)} = (d_1^{(k)}, d_2^{(k)}, \dots, d_n^{(k)})^\top,$$

and to be updated by this following algorithm

---

**Algorithm 4** Update of asymptotes

---

- 1: For iteration number  $k = 0, 1, \dots$  and constants  $L_{\min} < U_{\max}$ ,  $\xi \geq 0$ ,  $0 < T_1 < 1$  and
- 2:  $T_2 > 1$  we compute for each  $i = 1, \dots, n$

$$3: k < 2 : \quad L_i^{(k)} := x_i^{(k)} - \max \left\{ 1, |x_i^{(k)}| \right\}$$

$$U_i^{(k)} := x_i^{(k)} + \max \left\{ 1, |x_i^{(k)}| \right\}$$

If  $\text{sign} \left( x_i^{(k)} - x_i^{(k-1)} \right) \neq \text{sign} \left( x_i^{(k-1)} - x_i^{(k-2)} \right)$ , then

$$L_i^{(k)} := \max \left\{ x_i^{(k)} - \max \left\{ \xi, T_1 \left( x_i^{(k-1)} - L_i^{(k-1)} \right) \right\}, L_{\max} \right\}$$

$$4: k \geq 2 : \quad U_i^{(k)} := \min \left\{ x_i^{(k)} + \max \left\{ \xi, T_1 \left( U_i^{(k-1)} - x_i^{(k-1)} \right) \right\}, U_{\max} \right\}$$

else

$$L_i^{(k)} := \max \left\{ x_i^{(k)} - \max \left\{ \xi, T_2 \left( x_i^{(k-1)} - L_i^{(k-1)} \right) \right\}, L_{\min} \right\}$$

$$U_i^{(k)} := \min \left\{ x_i^{(k)} + \max \left\{ \xi, T_2 \left( U_i^{(k-1)} - x_i^{(k-1)} \right) \right\}, U_{\max} \right\}$$


---

The sub problems generated by these approximations can be solved explicitly, where the separability and the existence of an explicit solution of the functions  $f^{(k)}(x)$  and  $c_j^{(k)}(x)$ ,  $j = 1, \dots, m_c$  can be exploited.

In each iteration  $k$ , the asymptotes have to be adapted. The update rules are presented in the following algorithm,

where the value of  $d^{(k)}$  changes according to the convergence or divergence of the iterative process and the monotony of the objective function, for more information about this subject see ...,

---

**Algorithm 5** Update of asymptotes  $d^{(k)}$

---

1: For iteration  $k = 0, 1, \dots$  and constants  $L_{\min} < U_{\max}$ ,  $\xi \geq 0$ ,  $0 < C_1 < 1$  and  $C_2 > 1$  we compute for each  $i = 1, \dots, n$

2:

$$S_i^{(k)} := \begin{cases} -1, & \text{if } \frac{\partial f(x^{(k)})}{\partial x_i} < 0 \\ 1, & \text{if } \frac{\partial f(x^{(k)})}{\partial x_i} > 0 \end{cases}$$

3:

$$B_i^{(k)} := \begin{cases} S_i^{(k)} L_{\min}, & \text{if } S_i^{(k)} < 0 \\ S_i^{(k)} U_{\max}, & \text{if } S_i^{(k)} > 0 \end{cases}$$

4:

$k < 2$ :

$$d_i^{(k)} := x_i^{(k)} + S_i^{(k)} \max \{1, |x_i^{(k)}|\}$$

5:

$k \geq 2$  If  $\text{sign}(x_i^{(k)} - x_i^{(k-1)}) \neq \text{sign}(x_i^{(k-1)} - x_i^{(k-2)})$ , then

$$d_i^{(k)} := S_i^{(k)} \min \left\{ S_i^{(k)} (x_i^{(k)} + S_i^{(k)} \max \{ \xi, C_1 (x_i^{(k-1)} - d_i^{(k-1)}) \}), B_i^{(k)} \right\}$$

else

$$d_i^{(k)} := S_i^{(k)} \min \left\{ S_i^{(k)} (x_i^{(k)} + S_i^{(k)} \max \{ \xi, C_2 (x_i^{(k-1)} - d_i^{(k-1)}) \}), B_i^{(k)} \right\}$$


---

We can now formulate the main algorithm of our approximations according to the theory study on the previous sections

**Algorithm 6** NAMMA Method

- 
- 1: Choose starting point  $x^{(0)} \in IR^n$ . Set parameter  $\xi = 0, L_{\min} = -\infty, U_{\max} = \infty, C_2 > 1, 0 < C_1 < 1$ ,  
Compute  $f(x^{(0)}), g(x^{(0)})$ . Let  $k := 0$
  - 2: Stopping criteria.
  - 3: If  $x^{(k)}$  satisfies
  - 4: REPEAT
  - 5: Determine  $d_i^{(k)}, i = 1, \dots, n$ , by Algorithm 5 or let  $d^{(k)} = x^{(k)} + \sigma^{(k)}$  and take  $\sigma^{(k)}$  as defined in  
(4.59).
  - 6:  $x^{(k+1)} = d^{(k)} + S^{(k)}G^{(k)}$
  - 7: while Stopping criteria not satisfies.
- 

**4.5 Numerical examples**

In this section, we give some numerical result on NAMMA (new approach of the method of moving asymptotes). Our source code is written in Matlab 2017 using the Intel inside TM three core of the processor. Test problems are the standard unconstrained optimization problems and it will be found in [40]. The stopping criteria, we use are  $\|\nabla f(x^{(k)})\| < \varepsilon$ , where  $\varepsilon$  is a given tolerance, which depends on the precision of the final solution we want to find.

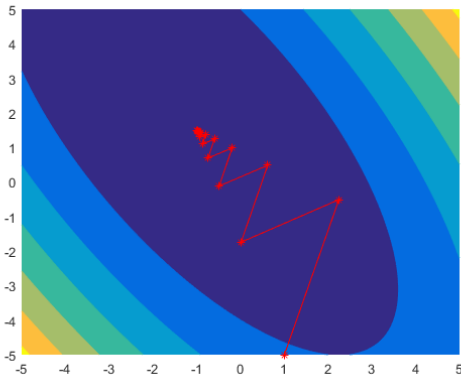
For the first examples the procedure was applied to the simple both dimensional quadratic functions

$$f_1(x_1, x_2) = x_1 - x_2 + 2x_1^2 + 2x_1x_2 + x_2^2,$$

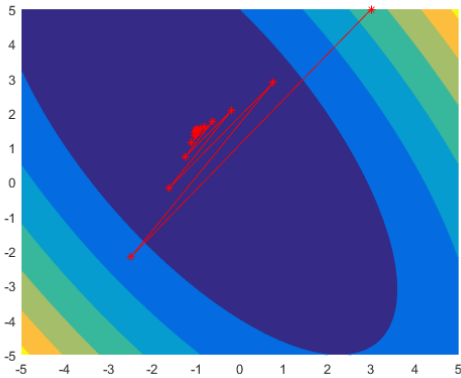
and

$$f_2(x_1, x_2) = (x_1 + 10x_2)^2 + 10x_1^4 + x_2^4$$

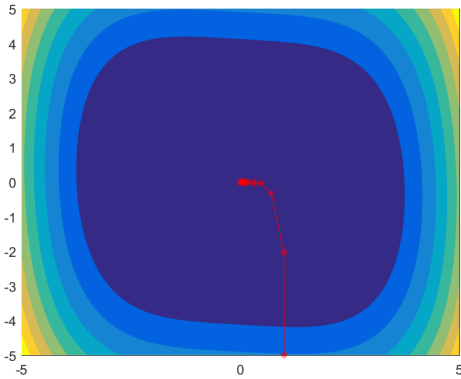
to know the behavior of the iterative sequence generated by the algorithm 5, and check numerically the theoretical results obtained in the previous sections.



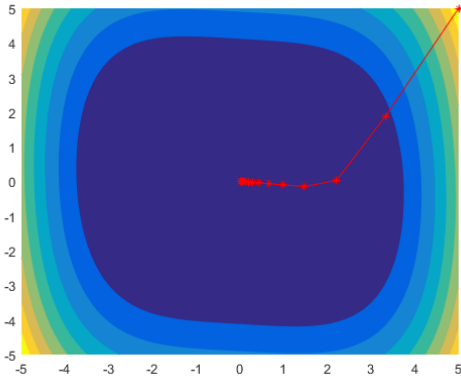
(a)



(b)



(c)



(d)

Figure 4.4 Convergence  $x^{(k)}$ ,  $X^{(k)}$  and  $Y^{(k)}$

$x^{(k)} = (x_1^{(k)}, x_2^{(k)})$	$\frac{\partial f}{\partial x}(x^{(k)})$	$\frac{\partial f}{\partial y}(x^{(k)})$
[-617630506282/225179981368, -279374430311/112589990684]	-14.9340	15.0000
[223078909527/225179981368, 182554721441/56294995342]	11.4483	-11.4483
[-842808070846/450359962737, -883902943720/1801439850948]	-7.4670	7.4670
[-538277928635/11529215046, 533993220509/225179981368]	5.7241	-5.7241
[-249707534978/225179981368, 281738333164/225179981368]	-3.7335	3.7335
[-788656779056/900719925474, 90574356241/56294995342]	2.8621	-2.8621
[-474887365299/450359962737, 309754251127/225179981368]	-1.8667	1.8667
[-28147390804/28147497671, 675541441004/450359962737]	1.4310	-1.4310
[-450359974431/450359962737, 337769958694/225179981368]	-0.9334	0.9334
[-900719898758/900719925474, 675539955799/450359962737]	0.7155	-0.7155
[-225179984292/225179981368, 675539930747/450359962737]	-0.4667	0.4667
[-225179978029/225179981368, 337769974976/225179981368]	0.3578	-0.3578
[-450359962528/450359962737, 337769972144/225179981368]	-0.2333	0.2333
[-112589990671/112589990684, 675539944151/450359962737]	0.1789	-0.1789
[-450359962759/450359962737, 675539944053/450359962737]	-0.1167	0.1167
[-450359962710/450359962737, 337769972064/225179981368]	0.0894	-0.0894
[-450359962748/450359962737, 84442493009/56294995342]	-0.0583	0.0583
[-112589990681/112589990684, 675539944116/450359962737]	0.0447	-0.0447
[-56294995342/56294995342, 168884986023/112589990684]	-0.0292	0.0292
[-225179981365/225179981368, 84442493013/56294995342]	0.0224	-0.0224
[-112589990684/112589990684, 337769972049/225179981368]	-0.0146	0.0146
[-450359962733/450359962737, 168884986027/112589990684]	0.0112	-0.0112
[-225179981369/225179981368, 675539944102/450359962737]	-0.0073	0.0073
[-900719925470/900719925474, 337769972053/225179981368]	0.0056	-0.0056
[-450359962737/450359962737, 675539944103/450359962737]	-0.0036	0.0036
[-900719925472/900719925474, 675539944106/450359962737]	0.0028	-0.0028
[-28147497671/28147497671, 84442493013/56294995342]	-0.0018	0.0018
[-112589990684/112589990684, 42221246506/28147497671]	0.0014	-0.0014
[-56294995342/56294995342, 168884986026/112589990684]	-0.0009	0.0009
[-225179981368/225179981368, 84442493013/56294995342]	0.0007	-0.0007
[-112589990684/112589990684, 337769972052/225179981368]	-0.0005	0.0005
[-450359962736/450359962737, 168884986026/112589990684]	0.0003	-0.0003
[-225179981368/225179981368, 675539944105/450359962737]	-0.0002	0.0002
[-900719925473/900719925474, 337769972052/225179981368]	0.0002	-0.0002
[-450359962737/450359962737, 675539944105/450359962737]	-0.0001	0.0001
[-900719925474/900719925474, 675539944105/450359962737]	0.0001	-0.0001
[-1, 337769972052/225179981368]	-0.0001	0.0001
[-450359962737/450359962737, 3/2]	0.0000	-0.0000
[-1, 3/2]	-0.0000	0.0000

Table 4.3 Convergence of  $x^{(k)}$

We can see from the figures (4.4a), (4.4b) and the table 4.3 that distance between two successive iterations converge to zero when we get closer to the solution. In the table 4.3, the set of values of  $x^{(k)}$  such that the derivatives of  $f$  at point  $x^{(k)}$  are positive represents the sub sequence  $X^{(k)}$  and those where the derivatives are negative represents the elements of the sub sequence  $Y^{(k)}$ . And the figures (4.4c) and (4.4d) represent respectively the convergence of  $X^{(k)}$  and  $Y^{(k)}$  for different starting point  $x^{(0)} = (1, -5)$  and  $x^{(0)} = (5, 5)$ , which shows a coherence between the numerical and theoretical result.

We have tested the new methods using the computational software Matlab 2017 with multiple-precision. We selected the following unimodal test functions  $f$ ,  $g$  and the multimodal function  $h$

The rosenbrock's function

$$f(\mathbf{x}) = \sum_{i=1}^{N-1} \left[ 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \right] \quad \text{where } \mathbf{x} = [x_1, \dots, x_N] \in \mathbb{R}^N,$$

the extended Woods's function

$$g(x) = \sum_{i=1}^N 100(x_{4i-3}^2 - x_{4i-2})^2 + (x_{4i-3} - 1)^2 + 90(x_{4i-1}^2 - x_{4i})^2 \\ + (1 - x_{4i-1})^2 + 10.1 \left\{ (x_{4i-2} - 1)^2 + (x_{4i} - 1)^2 \right\} + 19.8(x_{4i-2} - 1)(x_{4i} - 1),$$

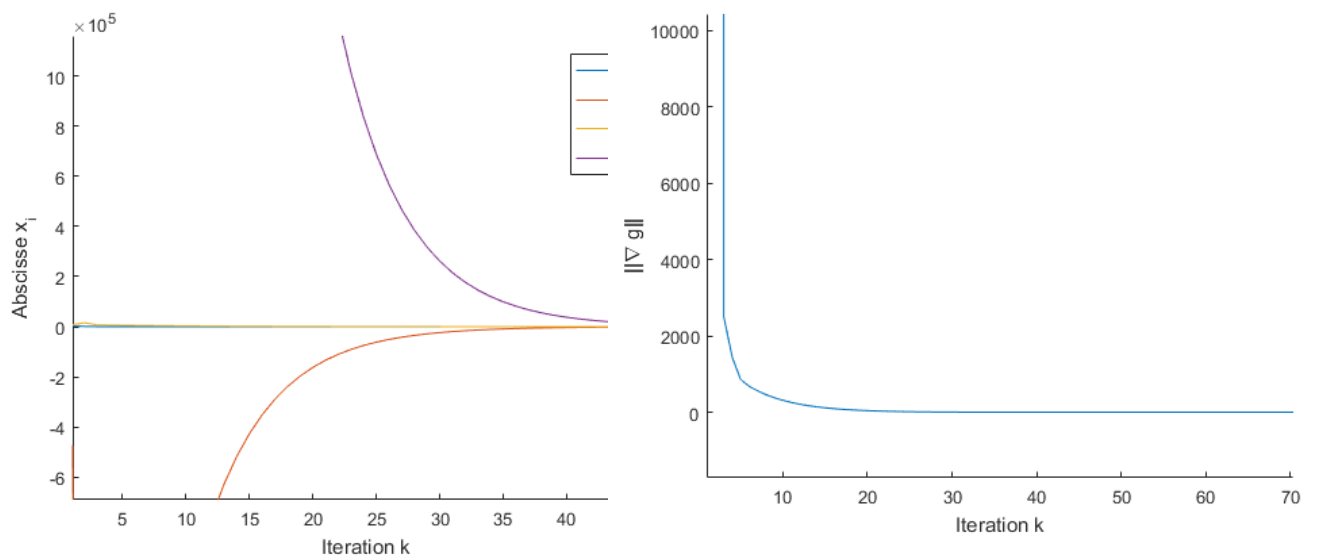
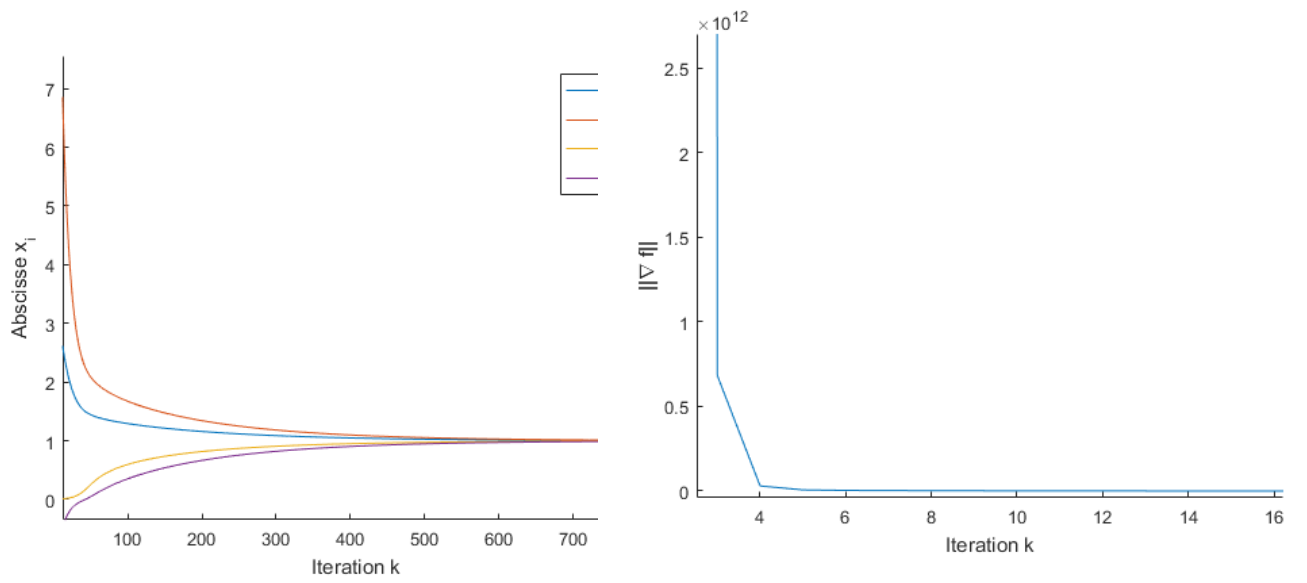
and extended Himmelblau function

$$h(x) = \sum_{i=1}^{n/2} (x_{2i-1}^2 + x_{2i} - 11)^2 + (x_{2i-1} + x_{2i}^2 - 7)^2$$

These functions has a minimums

Table 4.4 Numerical tests for non-convex non separable functions

function	$x_0$	Minimizer $x_*$	Minimum	CPU time(s)
$f$		$(1, 1, \dots, 1)$	$+ f_* = 0$	2D 0.001 3D 0.030
$g$		$(1, 1, \dots, 1)$	$g_* = 0$	2D 0.004 3D 0.020
$h$	$(3, 2)$	$(-2.80511808695, 3.13131251825)$	$h_* = 0$	0.051
	$(50, 35)$	$(3.584428340330, -1.84812652696)$		0.021
	$(2, 2.5)$	$(3.0, 2.0)$		0.025
	$(5, 3)$	$(-3.77931025337, -3.28318599128)$		0.007



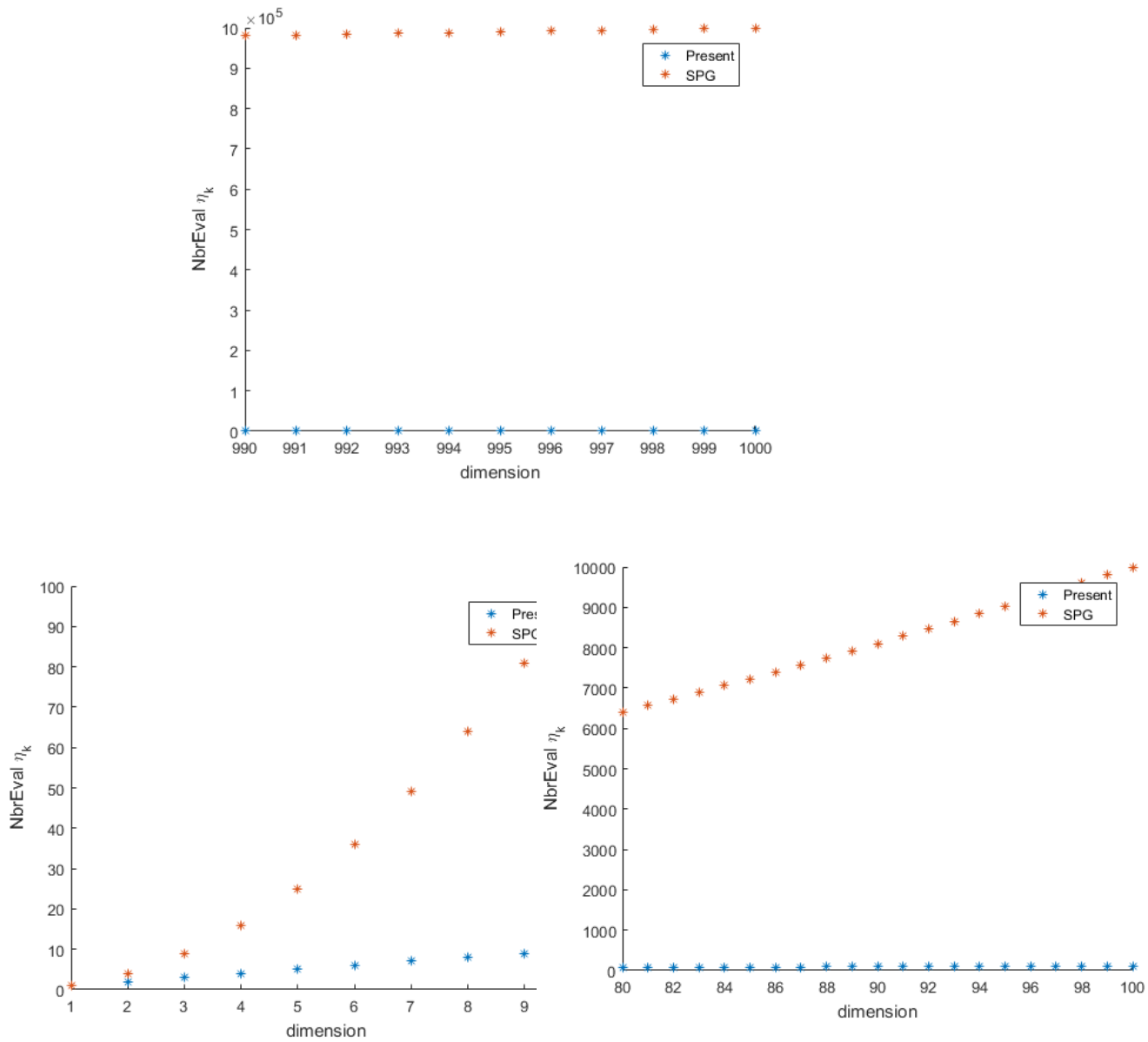


Figure 4.5 The evaluation's number of  $\eta^{(k)}$  for our algorithm and SPG method

Now we will test our algorithm on the following large scale optimization issues

**Problem 1**

$$f(x) = \sum_{i=1}^n f_i(x) \tag{4.107}$$

$$f_i(x) = x_i^2 \left( \frac{1}{2} + \frac{1}{3} x_i x_{n-2} x_{n-1} x_n \right) - x_i \left( 1 + \frac{1}{3} x_i^2 \right) \quad i = 1, \dots, n$$

**Problem 2**

$$f(x) = \sum_{i=1}^{n-1} f_i(x) + f_n(x) \tag{4.108}$$



$$\begin{aligned} f_i(x) &= \frac{1}{2}x_i^2 - \frac{0.1}{3}x_{i+1}^3 \quad i = 1, \dots, n-1 \\ f_n(x) &= \frac{1}{2}x_n^2 - \frac{0.1}{3}x_1^3 \end{aligned}$$

**Problem 3**

$$f(x) = f_1(x) + \sum_{i=2}^n f_i(x) \quad (4.109)$$

$$\begin{aligned} f_1(x) &= x_1 \left( \frac{1}{3}x_1^2 - \frac{3}{2}x_1 + 1 \right) + \sin(x_1 - x_2) \\ f_i(x) &= x_i \left( \frac{1}{3}x_i^2 - \frac{3}{2}x_i + 1 \right) + \sin(x_i - x_{i-1}) \quad i = 1, \dots, n \end{aligned}$$

**Problem 4**

$$f(x) = \sum_{i=1}^n f_i(x) \quad (4.110)$$

$$f_i(x) = x_i \sin x_i + \cos x_i - \frac{1}{2n}x_i^2 + x_n \cos x_i + x_i + \frac{1}{3}(x_i - 1)^3 + \frac{1}{2n} \sum_{i=1}^n x_i^2$$

$i=1, \dots, n$

**Problem 5**

$$f(x) = \sum_{i=1}^n f_i(x) \quad (4.111)$$

$$f_i(x) = \frac{1}{2}x_i^2 - (1 - x_i) \cos x_i + 0.99x_i^2 + 2x_i \quad i = 1, \dots, n$$

**Problem 6**

$$f(x) = \sum_{i=1}^n f_i(x) = \sum_{i=1}^{\frac{n}{3}} (f_{3i-2}(x) + f_{3i-1}(x) + f_{3i}(x)) \quad (4.112)$$

$$\begin{aligned} f_{3i-2}(x) &= (x_{3i} - 2x_{3i-1} - x_{3i}^2 - 1)x_{3i-2} \\ f_{3i-1}(x) &= \frac{1}{2}x_{3i-2}x_{3i-1}^2x_{3i} - x_{3i-2}^2x_{3i-1} + \frac{1}{3}x_{3i-1}^3 - 2x_{3i-1} \\ f_{3i}(x) &= x_{3i}(\exp -x_{3i-2} - \exp -x_{3i-1}) \end{aligned}$$

**Problem 7**

$$f(\mathbf{x}) = \sum_{i=1}^{n-1} \left[ 100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2 \right] \quad \text{where } \mathbf{x} = [x_1, \dots, x_n] \in \mathbb{R}^n, \quad (4.113)$$

Problem (P)	Present Algorithm					SPG		
	$x_0$	$n$	Iter	Time(s)	$\nabla f$	Iter	Time(s)	$\nabla f$
1	0.5e	10	6	0.0013199	3.04E-25	21	1.03457	2.03E-16
		500	8	0.0072169	-2.04E-20	30	7.86190	-7.15E-12
		5000	10	0.0268768	4.70E-18	24	33.4183	6.85E-11
		10000	14	0.0546120	-3.28E-16	43	49.4756	-2.44E-10
		100000	17	2.0378423	9.00E-15	55	72.8450	1.28E-12
		500000	21	4.0564789	-7.46E-15	64	164.784	-4.09E-10
2	e	10	7	0.0068820	8.06E-24	14	3.14153	1.09E-14
		1000	13	0.0093570	-7.64E-21	30	10.2148	-4.41E-15
		5000	18	0.0825634	1.71E-20	45	15.8130	8.50E-11
		100000	35	4.0231565	-3.28E-16	51	78.2764	-2.33E-16
		500000	47	10.45021	2.06E-15	100	134.415	6.41E-11
3	3e	100	24	0.005648	8.71E-20	20	7.14578	1.65E-14
		1000	37	0.015634	-2.92E-18	42	16.7610	3.84E-14
		5000	58	0.074512	6.74E-17	70	36.2194	5.72E-12
		100000	64	2.145623	-4.14E-14	121	84.1379	7.43E-15
		500000	75	8.045273	9.40E-16	213	140.051	1.24E-10
		5.10 <sup>7</sup>	97	24.05689	-8.50E-16			
4	0.01e	100	23	0.005123	1.65E-20	27	9.14523	1.65E-20
		500	31	0.015514	-3.84E-18	52	19.1876	-3.84E-18
		1000	46	2.045120	-7.29E-12	67	47.9437	-7.29E-12
		50000	61	4.004537	7.05E-15	124	67.1943	7.05E-15
		100000	53	6.003443	-9.40E-18	184	167.128	-9.40E-18
		5.10 <sup>7</sup>	76	12.85426	-6.72E-17			
5	e	100	7	0.008221	-1.23E-20	34	11.8134	-1.23E-20
		1000	9	0.018462	-9.40E-18	67	30.9172	-9.40E-18
		5000	11	0.723567	1.68E-12	91	58.4619	1.68E-12
		10000	19	4.123056	-6.53E-15	156	91.0791	-6.53E-15
		500000	22	9.043199	8.75E-18	238	205.417	8.75E-18
		5.10 <sup>7</sup>	68	23.12348	7.21E-17			
6	0.01e	50	10	0.045612	5.23E-20	19	5.19347	5.23E-20
		500	15	0.071348	8.75E-18	49	15.5673	8.75E-18
		10000	27	0.229835	-7.21E-12	64	32.3432	-7.21E-12
		50000	39	0.537564	1.68E-15	94	48.1892	1.68E-15
		500000	32	7.045739	-6.42E-18	125	184.643	-6.42E-18
		5.10 <sup>7</sup>	40	30.12348	3.28E-17			
7	10e	10	12	0.376138	9.06E-25	27	3.48216	9.06E-25
		100	30	0.285137	3.21E-20	45	14.7913	3.21E-20
		5000	16	8.234571	-5.91E-18	76	26.7800	-5.91E-18
		100000	34	25.56813	3.86E-18	85	40.6719	3.86E-18
		5.10 <sup>7</sup>	83	56.47267	-1.22E-16	278	277.160	-1.22E-16
	0.001e	10	15	0.376138	-8.50E-18	25	3.56412	-8.50E-18
		100	14	0.355246	6.78E-15	48	15.0046	6.78E-15
		5000	20	10.42577	1.87E-16	65	21.9715	1.87E-16
		100000	28	37.38117	7.87E-14	91	48.4912	7.87E-14
		5.10 <sup>7</sup>	101	84.49143	-5.91E-12	315	250.495	-5.91E-12

## 4.6 Conclusion

We described a general approach to the first order optimization based on optimizing the objective with respect to the updates. This gives an explicit optimization algorithm which can be viewed as an extension of the method of moving asymptotes for solving a large scale optimization problems, we showed that the proposed algorithm can converge more quickly than very known methods like Spectral Parameter Gradient method (SPG) and need more less evaluations of the spectral parameters thanks to the separability of our approximations, the other main contribution of this work is to show that we develop new algorithms for large optimization independently to the classical methods.

## Chapter 5

# NEW SEQUENTIAL CONVEX PROGRAMMING METHOD BASED ON THE MMA

### 5.1 Introduction

In this chapter we propose a new class of optimization method, this kind of methods can be belongs on the so called conservative convex separable approximation (CCSA) methods. These methods are intended for inequality constrained non linear programming problems. The new Sequential convex programming methods based on the MMA, proposed in this chapter, formulates separable and strictly convex non-linear subproblems iteratively by approximating the objective and the constraints. The asymptotes are introduced to truncate the feasible region. Due to the special structure, the resulting subproblems can be solved by appropriate methods, e.g., New approach of the method of moving asymptotes (NAMMA) presented in chapter 4. To ensure global convergence, a line search procedure is introduced. Moreover, an active set strategy is applied to reduce computation time.

The iterates of this new sequential programming methods are not always guaranteed to be inside the corresponding feasible region described by the constraints. As a consequence it is not able to solve some kind of optimization problems, for example the free material optimization problems as the compliance function is only well-defined on the feasible region of some of the constraints, for more information see [88, 55, 43]. We propose a modification of the resulting algorithm that ensures feasibility with respect to a given set of inequality constraints. The new procedure expands the resulting subproblems by additional nonlinear constraints, that are passed to the subproblem directly to ensure their feasibility in each iteration step. They are referred as feasibility constraints. In addition, other constraints may be violated within the optimization process. As globalization technique a line search procedure and an iterative algorithm for unconstrained optimization are used to ensure convergence. The resulting subproblems can be solved explicitly thanks to the Lagrangian function.

## 5.2 Position of the problem

We consider the general nonlinear optimization problem

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & c_j(x) \leq 0, \quad j = 1, \dots, m_c \\ & x_i^{\min} \leq x_i \leq x_i^{\max}, \quad i = 1, \dots, n, \end{aligned} \quad (5.1)$$

where  $f(x)$  and  $c_j(x)$ ,  $j = 1, \dots, m_c$ , are at least continuously differentiable on  $\mathbb{R}^n$ ,  $x_j^{\min}$  and  $x_j^{\max}$  are given real numbers such that  $x_j^{\min} < x_j^{\max}$  for each  $j$ .

Following Svanberg's approach, artificial variables  $y = (y_1, \dots, y_m)^T$  and  $z \in \mathbb{R}$  are introduced, so that the following enlarged problem is addressed:

$$\begin{aligned} \text{minimize} \quad & f(x) + \alpha_0 z + \sum_{i=1}^m \left( \beta_i y_i + \frac{1}{2} \gamma_i y_i^2 \right) \\ \text{subject to} \quad & c_i(x) - \alpha_i z - y_i \leq 0, \quad i = 1, \dots, m_c \\ & x_j^{\min} \leq x_j \leq x_j^{\max}, \quad j = 1, \dots, n \\ & y_i \geq 0, \quad i = 1, \dots, m_c \\ & z \geq 0 \end{aligned} \quad (5.2)$$

where  $\alpha_0, \alpha_i, \beta_i$  and  $\gamma_i$  are real numbers such that  $\alpha_0 > 0, \alpha_i \geq 0, \beta_i \geq 0, \gamma_i \geq 0$  and  $\beta_i + \gamma_i > 0$  for  $i = 1, \dots, m$ . Moreover,  $\alpha_i \beta_i > \alpha_0$  for all  $i$  such that  $\alpha_i > 0$ . The constants  $\beta_i$  must be chosen large enough so that the variables  $y_i$  are zero at the optimal solution. First we will talk about the so-called method of moving asymptotes (MMA), see [69–73]. MMA is a non linear programming approximation that creates a sequential of convex and separable sub problems, which are easy to solve due to their special structure. We solve the sub problems generated by these approximations and we use the resulting primal solution  $x^{(k)} \in \mathbb{R}^n$  in the corresponding iteration  $k$  to formulate a new sub problem. MMA achieves good results in practice, although no convergence proof is given. The idea behind the approximations presented in this chapter is the segmentation of the  $n$ -dimensional problem space into  $n$  one-dimensional spaces. One of the fundamental features is the introduction of the spectral parameters and a descent direction in the flexible asymptote  $d_i$ ,  $i = 1, \dots, n$ , for each optimization variable  $x_i$ ,  $i = 1, \dots, n$ , which truncate the feasible region. Important additional features are:

1. A rational approximations of the non linear inequality constraints and the objective function with respect to a flexible asymptotes  $d_i$ ,  $i = 1, \dots, n$  depending on a parameter  $\sigma_i$ ,  $i = 1, \dots, n$  at the current iteration point.
2. General algorithm applicable to any large and sparse non linear problem.
3. Generation of convex and separable sub problems has an explicit solution, i.e, diagonal Hessian matrices of the lagrangian function.
4. At each iteration, an explicit solution of the current sub problems generate a new sub problem close to the solution of original problem.

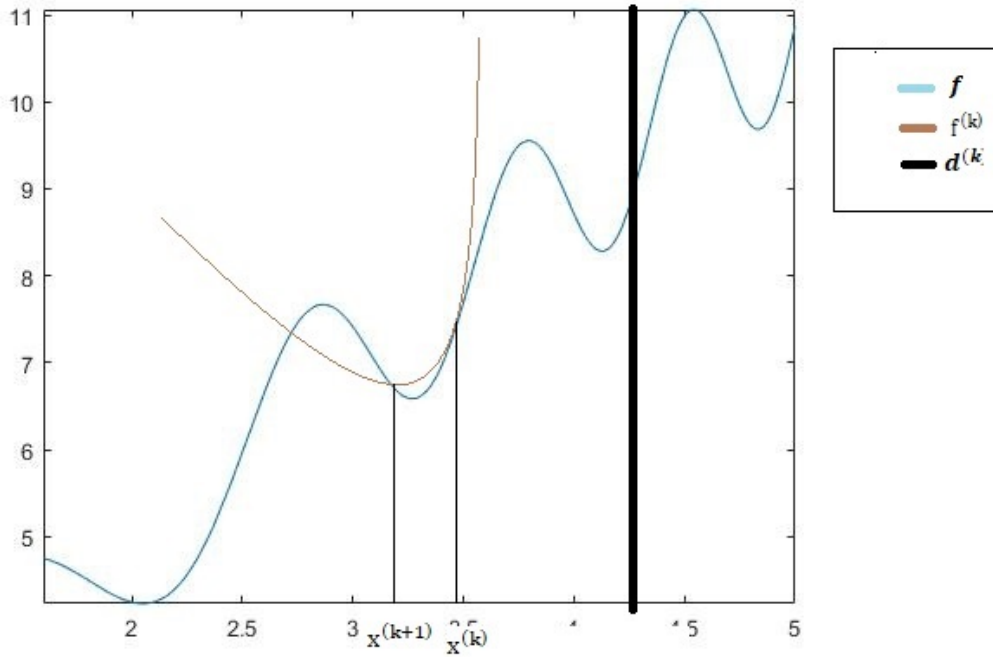


Figure 5.1 Scheme of convergence iterations

In each iteration  $k$  the objective function and the inequality constraints are linearized with respect to the inverse variables  $\frac{1}{x_i - d_i^{(k)}}$ . these approximation functions are defined on the set  $I^{(k)}$

$$I^{(k)} = I_1^{(k)} \times I_2^{(k)} \times \dots \times I_n^{(k)}, \quad (5.3)$$

where  $I_j^{(k)} = ]-\infty, d_j^{(k)}[ \cup ]d_j^{(k)}, +\infty[$ . Thus, the resulting approximation of the objective function at an iterate  $x^{(k)} \in \mathbb{R}^n$  is

$$\begin{aligned} f^{(k)}(x) := & f(x^{(k)}) \\ & + \sum_{i=1}^n \left[ \frac{\partial f(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\ & + \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)})(x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \end{aligned} \quad (5.4)$$

The non linear inequality constraints  $c_j(x)$ ,  $j = 1, \dots, m_c$ , are approximated analogously to 5.20 by

$$\begin{aligned} c_j^{(k)}(x) &:= c(x^{(k)}) \\ &+ \sum_{i=1}^n \left[ \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\ &+ \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)} (x_i - 2x_i^{(k)} + d_i^{(k)})) \right) \right] \end{aligned} \quad (5.5)$$

the vector  $\sigma^{(k)} = (\sigma_1^{(k)}, \dots, \sigma_n^{(k)})^T$  contains strictly positive parameters that are initially set and updated as stated later on in this chapter. and the moving asymptotes  $d^{(k)}$  are defined by:

$$d^{(k)} = x^{(k)} + \sigma^{(k)} \quad (5.6)$$

We obtain the subsequent sub problem by applying the approximations 5.20 and 5.21,

$$\begin{aligned} \min_x \quad & f^{(k)}(x) \\ \text{s.t.} \quad & c_j^{(k)}(x) \leq 0, \quad j = 1, \dots, m_c \\ & x_i \in I_i^{(k)}, \quad i = 1, \dots, n \end{aligned} \quad (5.7)$$

The functions are defined on the subset  $I^{(k)}$  given by 5.3.

The first and second derivatives of the convex approximations can be given analytically by

$$\frac{\partial f^{(k)}}{\partial x_i}(x) := \frac{\partial f(x^{(k)})}{\partial x_i} \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} + \frac{\sigma_i^{(k)}}{2} (x_i^{(k)} - d_i^{(k)}) \left( 1 - \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^2} \right) \quad (5.8)$$

$$\frac{\partial^2 f^{(k)}(x)}{\partial x_i \partial x_j} = 0, \quad \forall i \neq j \quad (5.9)$$

$$\frac{\partial^2 f^{(k)}}{\partial x_i^2}(x) := -\frac{\partial f(x^{(k)})}{\partial x_i} \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^3} + \sigma_i^{(k)} \frac{(x_i^{(k)} - d_i^{(k)})^3}{(x_i - d_i^{(k)})^3} \quad (5.10)$$

The derivatives for inequality constraints  $c_j^{(k)}(x)$ ,  $j = 1, \dots, m_c$ , can be obtained by replacing  $f(x)$  by  $c_j(x)$ ,  $j = 1, \dots, m_c$ .

$$\frac{\partial^2 c_j^{(k)}}{\partial x_i^2}(x) := -\frac{\partial c_j(x^{(k)})}{\partial x_i} \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^3} + \sigma_i^{(k)} \frac{(x_i^{(k)} - d_i^{(k)})^3}{(x_i - d_i^{(k)})^3} \quad (5.11)$$

It is easy to see that the functions are strictly convex:

$$\begin{aligned}
f^{(k)}(x^{(k)}) &= f(x^{(k)}), & c_j^{(k)}(x^{(k)}) &= c_j(x^{(k)}), \forall j = 1, \dots, m_c \\
\nabla f^{(k)}(x^{(k)}) &= \nabla f(x^{(k)}), & \nabla c_j^{(k)}(x^{(k)}) &= \nabla c_j(x^{(k)}), \forall j = 1, \dots, m_c \\
f^{(k)} &\text{ convex,} & c_j^{(k)} &\text{ convex, } \forall j = 1, \dots, m_c \\
f^{(k)} &\text{ separable,} & c_j^{(k)} &\text{ separable, } \forall j = 1, \dots, m_c
\end{aligned} \tag{5.12}$$

We introduce the terms  $\sigma_i^{(k)}$ ,  $\rho^{(k,l)}$  and  $\rho_i^{(k,l)}$ ,  $i = 1, \dots, m_c$  in the functions  $f^{(k,l)}$  and the constraints  $c_j^{(k,l)}$  aims to approximate the second order information by a spectral parameters to reduce the number of derivatives calculated during the algorithm generated by our approximation.

In the sequel a possible motivation for such terms is given, with the aim of analyzing the parameters  $\sigma_i^{(k)}$ ,  $\rho^{(k,l)}$  and  $\rho_i^{(k,l)}$  of the method. let us define the functions  $f^{(k,l)}$  and  $c_j^{(k,l)}$  by:

$$\begin{aligned}
f^{(k,l)}(x) &:= f(x^{(k)}) \\
&+ \sum_{i=1}^n \left[ \frac{\partial f(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\
&+ \frac{\rho^{(k,l)}}{2} \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + (x_i^{(k)} - d_i^{(k)})(x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \\
&= v(x, x^{(k)}, \sigma^{(k)}) + \rho^{(k,l)} w(x, x^{(k)}, \sigma^{(k)})
\end{aligned} \tag{5.13}$$

The non linear inequality constraints  $c_j(x)$ ,  $j = 1, \dots, m_c$ , are approximated analogously to 5.20 by

$$\begin{aligned}
c_j^{(k,l)}(x) &:= c_j(x^{(k)}) \\
&+ \sum_{i=1}^n \left[ \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\
&+ \frac{\rho_j^{(k,l)}}{2} \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + (x_i^{(k)} - d_i^{(k)})(x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \\
&= v_j(x, x^{(k)}, \sigma^{(k)}) + \rho_j^{(k,l)} w_j(x, x^{(k)}, \sigma^{(k)})
\end{aligned} \tag{5.14}$$

And then the approximating functions are chosen such that:

$$\begin{aligned}
f^{(k,l)}(x) &:= v^{(k)}(x) + \rho^{(k,l)} w^{(k)}(x) \\
c_j^{(k,l)}(x) &:= v_j^{(k)}(x) + \rho_j^{(k,l)} w_j^{(k)}(x)
\end{aligned} \tag{5.15}$$

The parameters  $\rho_i^{(k,l)}$  are strictly positive, and within an outer iteration  $k$ , the only difference between two inner iterations are the values of these parameters.



And then We obtain the new subsequent sub problem by applying the approximations 5.13 and 5.14,

$$\begin{aligned} \min_x \quad & f^{(k,l)}(x) + \alpha_0 z + \sum_{i=1}^m \left( \beta_i y_i + \frac{1}{2} \gamma_i y_i^2 \right) \\ \text{s.t.} \quad & c_j^{(k,l)}(x) - \alpha_i z - y_i \leq 0, \quad j = 1, \dots, m_c \\ & x_i \in I_i^{(k)}, \quad i = 1, \dots, n \end{aligned} \quad (5.16)$$

where  $\alpha_0, \alpha_i, \beta_i$  and  $\gamma_i$  are real numbers such that  $\alpha_0 > 0, \alpha_i \geq 0, \beta_i \geq 0, \gamma_i \geq 0$  and  $\beta_i + \gamma_i > 0$  for  $i = 1, \dots, m$ . Moreover,  $\alpha_i, \beta_i > \alpha_0$  for all  $i$  such that  $\alpha_i > 0$ . The constants  $\beta_i$  must be chosen large enough so that the variables  $y_i$  are zero at the optimal solution, in case the original problem has a nonempty feasible set. Problem (5.2) always has feasible points and at least an optimal solution, even if problem (5.1) has an empty feasible set; further, every local solution of problem (5.2) satisfies the Karush Kuhn-Tucker conditions, since the feasible set of problem (5.2) is qualified in the sense that it naturally fulfills a regularity condition. A CCSA method for solving problem (5.2) performs  $k$  outer and  $\ell$  inner iterations. The indices  $(k, \ell)$  are used to denote the  $\ell$ -th inner iteration within the  $k$ -th outer iteration.

Let consider the conservability condition

$$(C.C) : \begin{cases} f^{(k,l)}(\tilde{x}^{(k,l)}) \geq f(\tilde{x}^{(k,\ell)}), \\ c_j^{(k,l)}(\tilde{x}^{(k,l)}) \geq c_j(\tilde{x}^{(k,\ell)}), \quad \forall j \in \{0, 1, \dots, m_c\} \end{cases} \quad (5.17)$$

Since computing second derivatives of the objective function  $f$  and the constraints  $c_j$ ,  $j = 1, \dots, m_c$  is not possible or need an expensive time to calculate, for this reason we are approximated respectively by the scalars  $\rho_j^{(k,l)} \sigma_i^{(k)}$ , that is

$$\frac{\partial^2 f^{(k,l)}}{\partial x_i^2}(x) = -\frac{\partial f(x^{(k)})}{\partial x_i} \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^3} + \rho_j^{(k,l)} \sigma_i^{(k)} \frac{(x_i^{(k)} - d_i^{(k)})^3}{(x_i - d_i^{(k)})^3} \approx \frac{\partial^2 f(x)}{\partial x_j^2} \quad (5.18)$$

$$\frac{\partial^2 c_j^{(k,l)}}{\partial x_i^2}(x) = -\frac{\partial c_j(x^{(k)})}{\partial x_i} \frac{(x_i^{(k)} - d_i^{(k)})^2}{(x_i - d_i^{(k)})^3} + \rho_j^{(k,l)} \sigma_i^{(k)} \frac{(x_i^{(k)} - d_i^{(k)})^3}{(x_i - d_i^{(k)})^3} \approx \frac{\partial^2 c_j(x)}{\partial x_j^2} \quad (5.19)$$

### 5.3 Second order approximations

The approximations proposed in this chapter can be interpreted as a generalization of the method CONLIN, [23, 24], in which each approximation  $f^{(k)}$  and  $c_j^{(k)}$  is obtained by a linearization of the original functions in variables of the type  $\frac{1}{x_j - d_j}$ .

$$\begin{aligned} f^{(k)}(x) &:= f(x^{(k)}) \\ &+ \sum_{i=1}^n \left[ \frac{\partial f(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\ &+ \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \end{aligned} \quad (5.20)$$

$$\begin{aligned}
c_j^{(k)}(x) &:= c(x^{(k)}) \\
&+ \sum_{i=1}^n \left[ \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\
&+ \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)})(x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right]
\end{aligned} \tag{5.21}$$

The second derivative of the approximations  $f^{(k)}$  and  $c_j^{(k)}$  are given by (5.10) and (5.11). Since the approximations  $f^{(k)}$  and  $c_j^{(k)}$  are separable functions, its mixed second derivatives are zero. We suppose that the objective function  $f$  and the constraints inequality  $c_j$  are twice continuously differentiable, and we use in addition of 5.12:

$$\text{diag}(H_{f^{(k)}}(x^{(k)})) = \text{diag}(H_f(x^{(k)})) \tag{5.22}$$

$$\text{diag}(H_{c_j^{(k)}}(x^{(k)})) = \text{diag}(H_{c_j}(x^{(k)})) \tag{5.23}$$

The parameter  $\sigma^{(k)}$  are determined in such that the conditions 5.22 and 5.23 are satisfied.

Therefore by a simple computation, we verify that  $\sigma^{(k)}$  are explicitly given by:

$$\sigma^{(k)} = \text{diag}(H_f(x^{(k)})) \cdot e + 2\nabla f(x^{(k)}) \cdot R^{(k)}, \tag{5.24}$$

$$\sigma_j^{(k)} = \text{diag}(H_{c_j}(x^{(k)})) \cdot e + 2\nabla c_j(x^{(k)}) \cdot R^{(k)}, \tag{5.25}$$

## 5.4 Rules for updating the parameters $\rho_i^{(k,l)}$ , $\sigma_j^{(k)}$ and the asymptotes $d^{(k)}$ .

The updating of parameters  $\rho_i^{(k,l)}$  and  $\sigma_j^{(k)}$  discussed in the sequel are suggested in [32]. As far as the parameter  $\rho_i^{(k,l)}$  for  $l = 0$  the following values are used:

$$\begin{aligned}
\rho^{(1,0)} &= 1; & \rho^{(k+1,0)} &= \max \left\{ 0.1\rho^{(k,\tilde{l}(k))}, \rho^{\min} \right\}, \\
\rho_i^{(1,0)} &= 1; & \rho_i^{(k+1,0)} &= \max \left\{ 0.1\rho_i^{(k,\tilde{l}(k))}, \rho_i^{\min} \right\},
\end{aligned} \tag{5.26}$$

where  $\tilde{l}(k)$  is the number of inner iterations necessary to complete the  $k$ -th outer iteration and  $\rho_i^{\min}$  is a fixed strictly positive number. In each inner iteration, the updating of  $\rho_i^{(k,\ell)}$  is based on the solution of the latest sub problem. If  $c_j^{(k,l)}(\tilde{x}^{(k,l)}) < c_j(\tilde{x}^{(k,l)})$ , it is chosen  $\rho_i^{(k,l+1)}$  such that  $c_j^{(k,l+1)}(\tilde{x}^{(k,l)}) = c_j(\tilde{x}^{(k,l)})$  which gives  $\rho_i^{(k,\ell+1)} = \rho_i^{(k,\ell)} + \delta_i^{(k,\ell)}$  where

$$\begin{aligned}
\delta^{(k,\ell)} &= \frac{f(\tilde{x}^{(k,\ell)}) - f^{(k,\ell)}(\tilde{x}^{(k,\ell)})}{w^{(k)}(\tilde{x}^{(k,\ell)})} \\
\delta_i^{(k,\ell)} &= \frac{c_i(\tilde{x}^{(k,\ell)}) - c_i^{(k,\ell)}(\tilde{x}^{(k,\ell)})}{w_i^{(k)}(\tilde{x}^{(k,\ell)})}
\end{aligned} \tag{5.27}$$

Thus

$$\begin{aligned} \rho^{(k,\ell+1)} &= \min_{(k,\ell)} \left\{ 10\rho^{(k,\ell)}, 1.1 \left( \rho^{(k,\ell)} + \delta^{(k,\ell)} \right) \right\} & \text{if } \delta^{(k,\ell)} > 0 \\ \rho^{(k,\ell+1)} &= \rho^{(k,\ell)} & \text{if } \delta^{(k,\ell)} \leq 0 \end{aligned} \quad (5.28)$$

$$\begin{aligned} \rho_i^{(k,\ell+1)} &= \min_{(k,\ell)} \left\{ 10\rho_i^{(k,\ell)}, 1.1 \left( \rho_i^{(k,\ell)} + \delta_i^{(k,\ell)} \right) \right\} & \text{if } \delta_i^{(k,\ell)} > 0 \\ \rho_i^{(k,\ell+1)} &= \rho_i^{(k,\ell)} & \text{if } \delta_i^{(k,\ell)} \leq 0 \end{aligned} \quad (5.29)$$

The idea behind the updating of  $\rho_i$  is to increase or remain the same value of this parameter at each inner iteration, but never reduce it. Therefore, it is important that they decrease whenever an outer iteration starts to avoid slow convergence and generation of a small steps in the process. In terms of the parameters  $\sigma_j^{(k)}$ , the updating depends on the functions  $v_i^{(k)}$  and  $w_i^{(k)}$ . For this new sequential convex programming method, the Hessian matrix  $\nabla_{xx}^2 w_i^{(k)}(x)$  is diagonal with  $\frac{\partial^2 w_i^{(k)}}{\partial x_j^2}(x) \geq \frac{1}{\left(\sigma_j^{(k)}\right)^2}$  for all  $j$  and  $x$ . This means that the curvature of the function  $w_i^{(k)}$  towards  $x_j$  increases as  $\sigma_j^{(k)}$  decreases. Thus, depending on the pattern of the variables in the previous iterations, they should be stabilized or released, according to the following rule. if  $k = 1$  and  $k = 2$ :

$$\sigma_j^{(k)} = \frac{x_j^{\max} - x_j^{\min}}{2}, \quad (5.30)$$

and for  $k \geq 3$ :

$$\sigma_j^{(k)} = \gamma_j^{(k)} \sigma_j^{(k-1)}, \quad (5.31)$$

where

$$\gamma_j^{(k)} = \begin{cases} 0.7 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) < 0 \\ 1.2 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) > 0 \\ 1 & \text{if } \left( x_j^{(k)} - x_j^{(k-1)} \right) \left( x_j^{(k-1)} - x_j^{(k-2)} \right) = 0 \end{cases} \quad (5.32)$$

The sub problems generated by these approximations can be solved explicitly, where the separability and the existence of an explicit solution of the functions  $f^{(k)}(x)$  and  $c_j^{(k)}(x)$ ,  $j = 1, \dots, m_c$  can be exploited.

In each iteration  $k$ , the asymptotes have to be adapted. The update rules are presented in the following algorithm, where the value of  $d^{(k)}$  changes according to the convergence or divergence of the iterative process and the monotony of the objective function, for more information about this subject see ....

To update the parameters  $\rho_i$  between two consecutive inner iterations we adapt the approach given in [32]. To update  $\rho_i^{(k,l)}$  for the indices  $i = 1, \dots, n$  associated to the approximations (5.13) and (5.14) for which conservativity in  $x^{k,l}$  does not hold we proceed as follows:

**Algorithm 7** Update of parameters  $\rho_i^{(k,l)}$ 

- 
- 1: Compute  $s^{(k,\ell)} = \hat{x}^{(k,\ell)} - x^{(k)}$ , and for the indices  $i = 1, \dots, n$  associated to the approximations  $f^{(k,l)}(x)$  and  $c_j^{(k,l)}$  for which conservativity does not hold in  $x^{(k,l)}$ , compute
  - 2:  $y^{(k,\ell)} = \nabla f(\hat{x}^{(k,\ell)}) - \nabla f(x^{(k)})$ ;  $\eta^{(k,\ell)} = \frac{s^{(k,\ell)T} y^{(k,\ell)}}{s^{(k,\ell)T} s^{(k,\ell)}}$ ;  $\delta^{(k,\ell)} = \frac{f(\hat{x}^{(k,\ell)}) - f^{(k,\ell)}(\hat{x}^{(k,\ell)})}{w^{(k)}(\hat{x}^{(k,\ell)})}$
  - 3:  $y_j^{(k,\ell)} = \nabla c_j(\hat{x}^{(k,\ell)}) - \nabla c_j(x^{(k)})$ ;  $\eta_j^{(k,\ell)} = \frac{s^{(k,\ell)T} y_j^{(k,\ell)}}{s^{(k,\ell)T} s^{(k,\ell)}}$ ;  $\delta_j^{(k,\ell)} = \frac{c_j(\hat{x}^{(k,\ell)}) - c_j^{(k,\ell)}(\hat{x}^{(k,\ell)})}{w_j^{(k)}(\hat{x}^{(k,\ell)})}$
  - 4: If  $1.1(\rho_i^{(k,l)} + \delta_i^{(k,l)}) \leq 10\rho_i^{(k,l)}$  then set  $\rho_i^{(k,l+1)} = 1.1(\rho_i^{(k,l)} + \delta_i^{(k,l)})$
  - 5: Otherwise
  - 6:  $\rho_i^{(k,\ell)} < \eta_i^{(k,\ell)} < 1.1(\rho_i^{(k,\ell)} + \delta_i^{(k,\ell)})$  then set
  - 7:  $\rho_i^{(k,\ell+1)} = C_1(\eta_i^{(k,\ell)}) + C_2(1.1(\rho_i^{(k,\ell)} + \delta_i^{(k,\ell)}))$  with  $C_1 + C_2 = 1$ ,  $C_1, C_2 \geq 0$
  - 8: Now if  $\eta_i^{(k,\ell)} > 1.1(\rho_i^{(k,\ell)} + \delta_i^{(k,\ell)})$  then set:
  - 9:  $\rho_i^{(k,\ell+1)} = C_3(\eta_i^{(k,\ell)}) + C_4(1.1(\rho_i^{(k,\ell)} + \delta_i^{(k,\ell)}))$  with  $C_3 + C_4 = 1$ ,  $C_3, C_4 \geq 0$
  - 10: Finally if  $\eta_i^{(k,\ell)} \leq \rho_i^{(k,\ell)}$  then set  $\rho_i^{(k,\ell+1)} = 10\rho_i^{(k,\ell)}$ .
-

**Algorithm 8** Update of asymptotes  $d^{(k)}$ 

1: For iteration  $k = 0, 1, \dots$  and constants  $L_{\min} < U_{\max}$ ,  $\xi \geq 0$ ,  $0 < C_1 < 1$  and  $C_2 > 1$  we compute for each  $i = 1, \dots, n$

2:

$$S_i^{(k)} := \begin{cases} -1, & \text{if } \frac{\partial f(x^{(k)})}{\partial x_i} < 0 \\ 1, & \text{if } \frac{\partial f(x^{(k)})}{\partial x_i} > 0 \end{cases}$$

3:

$$B_i^{(k)} := \begin{cases} S_i^{(k)} L_{\min}, & \text{if } S_i^{(k)} < 0 \\ S_i^{(k)} U_{\max}, & \text{if } S_i^{(k)} > 0 \end{cases}$$

4:

$k < 2$ :

$$d_i^{(k)} := x_i^{(k)} + S_i^{(k)} \max \left\{ 1, |x_i^{(k)}| \right\}$$

5:

$k \geq 2$  If  $\text{sign}(x_i^{(k)} - x_i^{(k-1)}) \neq \text{sign}(x_i^{(k-1)} - x_i^{(k-2)})$ , then

$$d_i^{(k)} := S_i^{(k)} \min \left\{ S_i^{(k)} (x_i^{(k)} + S_i^{(k)} \max \{ \xi, C_1 (x_i^{(k-1)} - d_i^{(k-1)}) \}), B_i^{(k)} \right\}$$

else

$$d_i^{(k)} := S_i^{(k)} \min \left\{ S_i^{(k)} (x_i^{(k)} + S_i^{(k)} \max \{ \xi, C_2 (x_i^{(k-1)} - d_i^{(k-1)}) \}), B_i^{(k)} \right\}$$

For the choice of the constants  $C_1$  and  $C_2$ , Svanberg [72] and Zillober al [86] propose respectively the values  $(C_1, C_2) = (0.7, \frac{1}{C_2})$  and  $(C_1, C_2) = (0.7, 1.15)$ . Within the MMA procedure  $\xi = 0$  and  $U_{\max} = -L_{\min} = \infty$  and  $\xi > 0$ . These two constants are introduced in the the previous algorithm to distinguish two different situations. If  $\text{sign}(x_i^{(k)} - x_i^{(k-1)}) \neq \text{sign}(x_i^{(k-1)} - x_i^{(k-2)})$ , the distance between the asymptotes is reduced to prevent oscillation. As a consequence the domain shrinks. Otherwise, the distance is enlarged to allow larger steps and to speed up convergence. To start the algorithm we must initialize the parameters values of the lower and upper asymptotes. Svanberg [72] determines these values in the first iteration dependent on box constraints of the original problem. Now we can formulate the corresponding NSCP algorithm:

---

**Algorithm 9** NSCP algorithm
 

---

- 1: Choose starting point  $x^{(0)} \in \mathbb{R}^n$ . Set parameter  $\xi > 0$ ,  $U_{\max} = -L_{\min} = \infty$ ,  $C_2 > 1$ ,  $0 < C_1 < 1$ . Compute  $f(x^{(0)})$ ,  $\nabla f(x^{(0)})$ ,  $c_j(x^{(0)})$ ,  $\nabla c_j(x^{(0)})$ ,  $j = 1, \dots, m_c$ .
  - 2: Define  $\rho_i^{(1,0)}$  as in (5.26) for  $i = 1, \dots, m$ . Let  $k := 1$
  - 3: Stopping criteria: If  $x^{(k)}$  satisfies the KKT conditions of problem (5.2), stop and take  $x^{(k)}$  as a solution.
  - 4: Determine  $(\rho^{(k,l)}, \rho_i^{(k,l)})$  by algorithm (7).
  - 5: Set  $l = 0$
  - 6: If  $k > 1$ , compute  $\rho^{(k,l)}$  and  $\rho_i^{(k,l)}$  for  $i = 1, \dots, m$  as in (5.26) and  $\sigma^{(k)}$  as in (5.30)-(5.32).
  - 7: Compute  $d^{(k)} = x^{(k)} + \sigma^{(k)}$  (or we can compute  $d^{(k)}$  by algorithm (8)).
  - 8: Let  $f^{(k)}(x)$ ,  $c_j^{(k)}(x)$ ,  $j = 1, \dots, m_c$  be defined by (5.20) and (5.21).
  - 9: Generates the subproblem 5.7. Let  $x^{(k+1)}$  be the optimal solution of the sub-problem.
  - 10: Solves the sub-problem by Algorithm 6, obtaining  $\tilde{x}^{(k,l)}$ .
  - 11: Tests the conservativity condition
  - 12: If (5.17) does not hold for some index  $i = 1, \dots, m$
  - 13: Updating of the parameters  $\rho_i^{(k,l)}$ .
  - 14: Start a new inner iteration. Set  $l = l + 1$ . Go to step 9.
  - 15: Otherwise, set  $x^{(k+1)} = \tilde{x}^{(k,l)}$ ,  $k = k + 1$  and go to step 3.
- 

## 5.5 A New Sequential Convex Programming for constrained equality and/or inequality optimization

### 5.5.1 A new algorithm based on the SCP method of Zillober

The New Sequential Convex Programming for constrained equality and inequality optimization is an extension of the both methods MMA and NSCP. we proceed from the following optimization problem, where equality constraints are included additionally,

$$\begin{aligned}
 \min_x f(x) \quad & x \in \mathbb{R}^n \\
 \text{s.t.} \quad & c_j(x) = 0, \quad j = 1, \dots, m_e \\
 & c_j(x) \leq 0, \quad j = m_e + 1, \dots, m_c
 \end{aligned} \tag{5.33}$$

To ensure the convergence of our method to the exact solution of the optimization problem, let us introduce a merit function and the corresponding line search procedure, The merit function combines the objective function and the constraints in a suitable way.

we consider the augmented Lagrangian function  $\Phi_\rho : \mathbb{R}^{n+m_c} \rightarrow \mathbb{R}$  for a given set of penalty parameters

$\rho_j > 0, j = 1, \dots, m_c$

$$\Phi_\rho \begin{pmatrix} x \\ y \end{pmatrix} = f(x) + \sum_{j=1}^{m_c} \begin{cases} y_j c_j(x) + \frac{\rho_j}{2} c_j^2(x), & \text{if } j \in J(x) \\ -\frac{y_j^2}{2\rho_j}, & \text{otherwise} \end{cases} \quad (5.34)$$

where  $y \in \mathbb{R}^{m_c}$  are the corresponding Lagrangian multipliers. Moreover, we define the active set respect to the augmented Lagrangian by  $J(x)$  and its complement by  $\bar{J}(x)$ ,

$$\begin{aligned} J(x) &:= \{1 \leq j \leq m_e\} \\ &\cup \left\{ m_e + 1 \leq j \leq m_c \mid -\frac{y_j}{\rho_j} \leq c_j(x) \right\} \\ \bar{J}(x) &:= \{m_e + 1 \leq j \leq m_c \mid j \notin J\} \end{aligned} \quad (5.35)$$

It can be check easily that  $\Phi_\rho \begin{pmatrix} x \\ y \end{pmatrix}$  is differentiable. The penalty parameters  $\rho \in \mathbb{R}^{m_c}$  must be carefully adapted during the solution process to guarantee a sufficient descent and global convergence, see [60]. Choosing the augmented Lagrangian merit function is motivated by the following properties, see [84]:

**Definition 5.5.1** *The linear independence constraint qualification (LICQ) is satisfied at a feasible solution  $x \in \mathbb{F}$ , if the gradients of the active constraints are linearly independent at  $x$ .*

**Lemma 5.5.2** • *A point  $(x^*, y^*)$  is stationary for  $\Phi_\rho \begin{pmatrix} x \\ y \end{pmatrix}$  defined by 5.34 for a positive fixed  $\rho \in \mathbb{R}^{m_c}$ , if and only if it is stationary for problem 5.33.*

- *Let  $(x^*, y^*)$  be stationary for problem 5.33 and let the gradients of the active constraints be linearly independent in  $x^*$ , i.e., Definition 5.5.1 holds. Then there exists a positive parameter in  $x^*$  is a local minimizer for  $\Phi_\rho \begin{pmatrix} x \\ y^* \end{pmatrix}, \forall \rho \geq \rho^*$*

**Proof 13** *See Fletcher [19, 20]*

To ensure strict convexity of the approximated objective function  $f^{(k)}(x)$  and thus an unique solution of the subproblem, we suppose that the parameters  $\sigma^{(k)}$  and  $\rho^{(k,l)}$  are strictly positive to get respectively  $\frac{\partial^2 f^{(k)}(x)}{\partial^2 x_i} > 0$  and  $\frac{\partial^2 f^{(k,l)}(x)}{\partial^2 x_i} > 0$  holds for all  $i = 1, \dots, n$ . and we compute at the value of  $\frac{\partial^2 f^{(k)}(x)}{\partial^2 x_i} > 0$  at  $x^{(k)}$  in 5.10 we find:

$$\frac{\partial^2 f^{(k)}}{\partial x_i^2}(x^{(k)}) := \sigma_i^{(k)} - \frac{\frac{\partial f(x^{(k)})}{\partial x_i}}{(x_i^{(k)} - d_i^{(k)})} > 0 \quad (5.36)$$

$$\sigma_i^{(k)} > \frac{\frac{\partial f(x^{(k)})}{\partial x_i}}{(x_i^{(k)} - d_i^{(k)})} \quad (5.37)$$

The algorithm 5 ensure that  $\frac{\partial f(x^{(k)})}{\partial x_i}$  and  $x_i^{(k)} - d_i^{(k)}$  has the same sign for all  $i = 1, \dots, n$ , then the right quotient of inequality 5.38 is always positive and the inequality is true. Doing the same computation for

inequality constraints, we find:

$$\sigma_{i,j}^{(k)} > \frac{\frac{\partial c_j(x^{(k)})}{\partial x_i}}{(x_i^{(k)} - d_i^{(k)})} \quad j = m_e + 1, \dots, m_c. \quad (5.38)$$

And the equality constraints  $c_j(x), j = 1, \dots, m_e$ , are linearized by:

$$c_j^{(k)}(x) := c_j(x^{(k)}) + \sum_{i=1}^n \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i - x_i^{(k)}). \quad (5.39)$$

The objective function  $f$  and the inequality constraints  $c_j(x), j = m_e + 1, \dots, m_c$  are approximated respectively by 5.20 and 5.21. The first and second order derivatives of the convex approximations can be given analytically by 5.10 and 5.11. The corresponding subproblem is formulated by

$$\begin{aligned} \min_x & f^{(k)}(x) \\ \text{s.t.} & c_j^{(k)}(x) = 0, \quad j = 1, \dots, m_e \\ & c_j^{(k)}(x) \leq 0, \quad j = m_e + 1, \dots, m_c \\ & x_i \in I_i^{(k)}, \quad i = 1, \dots, n \end{aligned} \quad (5.40)$$

Due to strict convexity of the approximations of the resulting subproblem 5.40 possesses an unique solution. We denote by the primal solution of subproblem 5.40 in iteration  $k$  by  $z^{(k)} \in \mathbb{R}^n$  and the dual solution by  $v^{(k)} \in \mathbb{R}^{m_c}$ . In each iteration, it is tested yields a sufficient descent with respect to the augmented Lagrangian merit function. Therefore, the so-called Armijo steplength algorithm, see Armijo [2] is applied. In each iteration  $k$ , the stepsize  $\gamma_{(k,p)}$ , with  $\gamma^{(k,0)} := 1$ , is reduced by a constant factor  $\alpha \in (0, 1)$  iteratively, i.e.,

$$\gamma^{(k,i+1)} := \alpha \gamma^{(k,i)} \quad (5.41)$$

until the following condition is satisfied for the first time

$$\Phi_{\rho^{(k)}} \left( \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \sigma^{(k,i)} d^{(k)} \right) \leq \Phi_{\rho^{(k)}} \left( \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} \right) + r \sigma^{(k,i)} \nabla \Phi_{\rho^{(k)}} \left( \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} \right)^T d^{(k)}, \quad (5.42)$$

where  $r \in (0, 1)$  is constant and where the search direction  $s_d^{(k)} \in \mathbb{R}^{n+m_c}$  is given by

$$s_d^{(k)} := \begin{pmatrix} z^{(k)} - x^{(k)} \\ v^{(k)} - y^{(k)} \end{pmatrix}. \quad (5.43)$$

Condition 5.42 ensures a sufficient descent in the augmented Lagrangian merit function. To update the penalty parameter  $\rho_i^{(k)}, i = 1, \dots, m_c$ , additional parameters  $\eta_i^{(k)}, i = 1, \dots, n$ , are introduced, with estimate the curvature of the approximated function  $f^{(k)}(x)$ . From the Mean value Theorem of the integral calculus we know that, given a continuously differentiable function  $f: \mathbb{R}^n \rightarrow \mathbb{R}$  we have that  $f(y) = f(x) + \nabla f(x + \alpha(y-x))^T (y-x)$ , for some  $\alpha \in (0, 1)$ . Moreover, if  $f$  is twice continuously differentiable, then:

$$\nabla f(y) = \nabla f(x) + \int_0^1 \nabla^2 f(x + \alpha(y-x)) d\alpha (y-x) \quad (5.44)$$



By setting  $s = y - x$ , the scalar

$$\eta = \frac{s^T t}{s^T s}, \tag{5.45}$$

where  $t = \nabla f(y) - \nabla f(x)$ , or again by the Mean Value Theorem  $t = \left( \int_0^1 \nabla^2 f(x + \alpha s) d\alpha \right) s$ , defines a Rayleigh quotient with respect to the average Hessian matrix  $\left( \int_0^1 \nabla^2 f(x + \alpha s) d\alpha \right)$ . Such quotient has its value between the smallest and the largest eigenvalue of the average Hessian matrix, what motivates the terminology spectral parameter for 5.45. Thus, if we require that the Hessian of the functions  $c_i$  and  $f$  are approximated by scalar matrices, we might say that  $\eta I$  is the matrix of such type that best approximates the average Hessian. The idea is to replace the diagonal of the Hessian matrix  $diag(H_f^{(k)})$  and  $diag(H_{c_j}^{(k)})$  by the spectral parameter 5.45

$$\nabla^2 f(x) \approx \eta_0 I \Rightarrow \frac{\partial^2 f(x)}{\partial x_j^2} \approx \eta_0 \tag{5.46}$$

$$\nabla^2 c_j(x) \approx \eta_j I \Rightarrow \frac{\partial^2 c_j(x)}{\partial x_i^2} \approx \eta_j \tag{5.47}$$

Such approximation will be used in the iterative process of our method if conservativity 5.17 does not hold for some approximating function  $f^{(k)}$  or  $c_j^{(k)}$  in  $\tilde{x}^{(k,\ell)}$ . In such case, the corresponding function is modified with an increased parameter  $\rho_i$ , that is, by computing  $\rho_i^{(k,\ell+1)}$  such that  $\rho_i^{(k,\ell+1)} > \rho_i^{(k,\ell)}$ . The second order information contained in the spectral parameter is then used to obtain  $\rho_i^{(k,\ell+1)}$ .

The points used to compute the direction  $s$  are the current estimate  $x^{(k)}$  and the solution of the latest subproblem  $\tilde{x}^{(k,\ell)}$ , that is,  $s^{(k,\ell)} = \tilde{x}^{(k,\ell)} - x^{(k)}$ . It is also necessary to compute vector  $t_i^{(k,\ell)} = \nabla f_i(\tilde{x}^{(k,\ell)}) - \nabla f_i(x^{(k)})$  for the indices  $i \in \{0, 1, \dots, m\}$  associated to the approximations  $f^{(k)}$  or  $c_j^{(k)}$  that do not verify condition 5.17, so that:

$$\eta_j^{(k,\ell)} = \frac{(s^{(k,\ell)})^T t_i^{(k,\ell)}}{(s^{(k,\ell)})^T (s^{(k,\ell)})} \quad j = 0, \dots, m_c \tag{5.48}$$

The penalty parameters are updated until the descent property

$$\nabla \Phi_{\rho^{(k)}} \left( \begin{matrix} x^{(k)} \\ y^{(k)} \end{matrix} \right)^T d^{(k)} \leq - \frac{\eta^{(k)} (\delta^{(k)})^2}{2} \tag{5.49}$$

is satisfied, where  $\hat{\delta}^{(k)} \in \mathbb{R}$  denotes the norm of the search direction with respect to the primal variable  $x^{(k)}$ , i.e.,

$$\delta^{(k)} := \left\| z^{(k)} - x^{(k)} \right\|_2 \tag{5.50}$$

Within the update loop the penalty parameters are denoted by  $\hat{\rho}_j^{k,i}$   $j = 1, \dots, m_c$ , where  $i$  denotes the  $i$ -th penalty parameter update within iteration  $k$ .  $\hat{\rho}_j^{k,0}$  is initialized by  $\hat{\rho}_j^{k-1}$   $j = 1, \dots, m_c$ . The corresponding update is described by Algorithm 10 according to Zillober [85].

**Algorithm 10** Update of the penalty parameters

- 
- 1: Let  $c_1 > 1$ ,  $c_2 > c_1$  be suitable constants. Let  $x^{(k)} \in \mathbb{R}^n$  and  $y^{(k)} \in \mathbb{R}^{m_c}$  be respectively the current primal and dual variables. Moreover,  $(z^{(k)}, v^{(k)})$  is the solution of subproblem (5.40) defined in  $x^{(k)}$  and  $\hat{\rho}_j^{(k,i)}$ ,  $j = 1, \dots, m_e$ , is a given penalty parameter.
  - 2: **if**  $j \in \{1 \leq j \leq m_e\}$  or  $j \in \left\{ m_e + 1 \leq j \leq m_c \mid -\frac{y_j^{(k)}}{\rho_j^{(k,i)}} \leq c_j(x^{(k)}) \right\}$  **then**
  - 3:   **if**  $c_j(x^{(k)}) > 0$  and  $\nabla c_j(x^{(k)})^T (z^{(k)} - x^{(k)}) \neq 0$  or  $c_j(x^{(k)}) < 0$  and  $\nabla c_j(x^{(k)})^T (z^{(k)} - x^{(k)}) > 0$  **then**
  - 4:     
$$\rho_j^{(k,i+1)} := \min \left\{ \kappa_2 \rho_j^{(k,i)}, \max \left\{ \kappa_1 \rho_j^{(k,i)}, \left| \frac{2(v_j^{(k)} - y_j^{(k)})}{c_j(x^{(k)})} \right| \right\} \right\}$$
  - 5:   **else**
  - 6:     
$$\rho_j^{(k,t+1)} := \kappa_1 \rho_j^{(k,i)}$$
  - 7:
  - 8:   **if**  $(v_j^{(k)} - y_j^{(k)}) < 0$  **then**
  - 9:     
$$\rho_j^{(k,t+1)} := \min \left\{ \kappa_2 \rho_j^{(k,l)}, \max \left\{ \kappa_1 \rho_j^{(k,l)}, \left| \frac{y_j^{(k)} (v_j^{(k)} - y_j^{(k)}) 4m_c}{\eta^{(k)} (\delta^{(k)})^2} \right| \right\} \right\}$$
  - 10:   **else**
  - 11:     
$$\rho_j^{(k,l+1)} := \kappa_1 \rho_j^{(k,l)}$$
- 

The constants  $c_1$  and  $c_2$  respectively prevents that the penalty parameters converge too slowly and ensures that these last one do not increase too quickly. Zillober [82] proposes to set  $c_1 = 2$  and  $c_2 = 10$ .

To prove global convergence we update the asymptotes  $d^{(k)}$  as in the chapter 3. Zillober [82] proposed an algorithms to adapt carefully a lower and upper asymptotes  $L^{(k)}$  and  $U^{(k)}$  under certain conditions. The asymptotes need to be feasible according to the definition in chapter 3 to ensure convergence of the new sequential convex programming (NSCP) methods for the optimization problems with equality and inequality constraints. In general we can choose different asymptotes for the objective function and each constraint as in the algorithms 2-6 to improve the performance of the algorithm, even the computational effort is much higher, otherwise it is possible to update them according to algorithm 8. There are also many authors proposed different algorithms to compute iteratively these asymptotes.

The main algorithm to solve the problem (5.33) can be introduced as:

**Algorithm 11**

- 
- 1: Choose starting point  $x^{(0)} \in \mathbb{R}^n$  and  $y^{(0)}$ . Compute  $f(x^{(0)}), \nabla f(x^{(0)}), c_j(x^{(0)}), \nabla c_j(x^{(0)}), j = 1, \dots, m_c$ . Set parameters  $\omega \in ]0; 1[, r \in (0, 1), \beta \in (0, 1), \tau > 0, c_2 > c_1 > 1$  and penalty parameters  $\rho_j^{(-1)} > 0, j = 1, \dots, m_c$ . Let  $k := 0$
  - 2: Determine  $d^{(k)}$  for the objective and constraints as in Algorithm 6 or by Algorithm 8. Let  $f^{(k)}(x), c_j^{(k)}(x), j = 1, \dots, m_c$  be defined by (5.20) and (5.21). Formulate (5.40) for the corresponding iteration  $k$ .
  - 3: Solve (5.40). Let  $z^{(k)}$  be the optimal solution of (5.40) and  $v^{(k)}$  the vector of the corresponding Lagrangian multipliers.
  - 4: **If**  $z^{(k)} = x^{(k)}$ , then STOP.  $(x^{(k)}, v^{(k)})$  is a KKT point of (5.33).
  - 5: Let  $s_d^{(k)} = \begin{pmatrix} z^{(k)} - x^{(k)} \\ v^{(k)} - y^{(k)} \end{pmatrix}$ ,  $\hat{\delta}^{(k)} := \|z^{(k)} - x^{(k)}\|_2$  and  $\eta^{(k)}$  as defined in (4.6). Let  $i = 0$  and  $\rho^{(k,0)} := \rho^{(k-1)}$ .
  - 6: Compute  $\Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}^T s_d^{(k)}$
  - 7: **If**  $\nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}^T s_d^{(k)}$ , update penalty parameters according to Algorithm 10. Let  $i = i + 1$  and go to Step 6. Otherwise, let  $\rho^{(k)} := \rho^{(k,i)}, i = 0$  and  $\sigma^{(k,0)} := 1$
  - 8: Compute  $f(x^{(k)} + \sigma^{(k,i)}(z^{(k)} - x^{(k)})), c_j(x^{(k)} + \sigma^{(k,i)}(z^{(k)} - x^{(k)})), j = 1, \dots, m_c$ , and  $\Phi_{\rho^{(k)}} \left( \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \sigma^{(k,i)} d^{(k)} \right)$ .
  - 9: **if** (5.42) is not satisfied **then** let  $\sigma^{(k,i+1)} := \beta \sigma^{(k,i)}, i = i + 1$  and repeat (Armijo). Otherwise,  $\sigma^{(k)} := \sigma^{(k,i)}$ .
  - 10: Let  $\begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} := \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \sigma^{(k)} s_d^{(k)}, k := k + 1$ .
  - 11: Compute  $\nabla f(x^{(k)}), \nabla c_j(x^{(k)}), j = 1, \dots, m_c$  and go to Step 1.
- 

Sub problem (5.40) can be solved by the algorithm 9 or by an interior point method, see [1, 33, 53, 54]. And according to [83], we can reduce the size of the primal-dual system of linear equations if the number of constraints and variables is too large. For more information about the convergence proof, the choice of the asymptotes to control the curvature of the merit function, and some numerical results for problems resulting from topology optimization, see [49]

## 5.6 A feasible sequential convex programming based on the NAMMA

In many applications, we get the optimization problem's domain specified by other constraints. Since most nonlinear optimization methods cannot ensure feasibility during the solution process, these problems cannot

solved appropriately. For example if we get

$$\begin{aligned} c_1(x) &:= \log(e_1(x)) \\ c_2(x) &:= \sqrt{e_2(x)} \end{aligned} \quad (5.51)$$

where  $e_1(x)$  and  $e_2(x)$  are nonlinear functions. To ensure that  $c_1(x)$  and  $c_2(x)$  can be evaluated, the constraints

$$\begin{aligned} e_1(x) &> 0, \\ e_2(x) &> 0, \end{aligned} \quad (5.52)$$

need to be satisfied. Note that we require  $e_1(x) > 0$  and  $e_2(x) > 0$  such that  $c_1(x)$  and  $c_2(x)$  are continuously differentiable for all  $x$  satisfying (5.51).

To guarantee feasibility of a given subset of constraints in each iteration, we present an extended version of the algorithm 12, which the resulting algorithm assumed that the constraints  $c_j(x)$ ,  $j = 1, \dots, m_c$  and the objective function  $f(x)$  can only be evaluated, if all feasibility constraints  $e_j(x) \leq 0$ ,  $j = 1, \dots, m_f$ , are satisfied.

We extend the problem (5.33) by an additional feasibility constraints  $e_j(x)$ ,  $j = 1, \dots, m_f$ . We get

$$\begin{aligned} \min_x \quad & f(x) \quad x \in \mathbb{R}^n \\ \text{s.t.} \quad & c_j(x) = 0, \quad j = 1, \dots, m_e \\ & c_j(x) \leq 0, \quad j = m_e + 1, \dots, m_c \\ & e_j(x) \leq 0, \quad j = 1, \dots, m_f \end{aligned} \quad (5.53)$$

The objective function  $f(x)$  and constraints  $c_j(x)$ ,  $j = 1, \dots, m_c$  are supposed to be at least continuously differentiable on the subset

$$F := \{x \in \mathbb{R}^n \mid e_j(x) \leq 0, j = 1, \dots, m_f\} \quad (5.54)$$

The functions  $e_j(x)$ ,  $j = 1, \dots, m_f$ , must be convex and at least twice continuously differentiable on  $\mathbb{R}^n$ , for more information about this subject, see [46]. As a consequence,  $F$  is convex, these last condition is important to guarantee domain's feasibility, when the stepsize is reduced during a line search procedure.

Let us start the procedure by taking a feasible point  $x^{(0)} \in F$ , as the previous algorithms, this one generates a sequence of convex subproblems, which are easy to solve explicitly due to their special structure. Moreover, we need to check at each iteration  $k$ , if the nonlinear constraints are satisfied to ensure their feasibility. The objective function  $f$  and the constraints  $c_j$ ,  $j = 1, \dots, m_c$ , are approximated by convex and separable functions according to (5.20), (5.21) denoted by  $f^{(k)}$  and  $c_j^{(k)}$ ,  $j = 1, \dots, m_c$ , on the other hand we keep the constraints  $e_j(x)$ ,  $j = 1, \dots, m_f$  in the generated subproblems. We can assume that the functions  $e_1(x), \dots, e_{m_f}(x)$  and their derivatives are much easier to evaluate than the other constraints, objective and their gradients.

The resulting subproblems at an iteration  $k$

$$\begin{aligned} \min_x \quad & f^{(k)}(x) \quad x \in \mathbb{R}^n \\ \text{s.t.} \quad & c_j^{(k)}(x) = 0, \quad j = 1, \dots, m_e \\ & c_j^{(k)}(x) \leq 0, \quad j = m_e + 1, \dots, m_c \\ & e_j(x) \leq 0, \quad j = 1, \dots, m_f \\ & x_i \in I_i^{(k)} \end{aligned} \quad (5.55)$$

The approximations are defined by

$$\begin{aligned} f^{(k)}(x) &:= f(x^{(k)}) \\ &+ \sum_{i=1}^n \left[ \frac{\partial f(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\ &+ \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \end{aligned}$$

for  $j = 1, \dots, m_e$

$$c_j^{(k)}(x) := c_j(x^{(k)}) + \sum_{i=1}^n \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i - x_i^{(k)}).$$

for  $j = m_e + 1, \dots, m_c$

$$\begin{aligned} c_j^{(k)}(x) &:= c(x^{(k)}) \\ &+ \sum_{i=1}^n \left[ \frac{\partial c_j(x^{(k)})}{\partial x_i} (x_i^{(k)} - d_i^{(k)})^2 \left( \frac{1}{x_i^{(k)} - d_i^{(k)}} - \frac{1}{x_i - d_i^{(k)}} \right) \right] \\ &+ \sum_{i=1}^n \left[ \sigma_i^{(k)} \left( \frac{1}{2} \frac{(x_i^{(k)} - d_i^{(k)})^3}{x_i - d_i^{(k)}} + \frac{1}{2} (x_i^{(k)} - d_i^{(k)}) (x_i - 2x_i^{(k)} + d_i^{(k)}) \right) \right] \end{aligned}$$

The asymptotes  $d_i^{(k)}$ ,  $i = 1, \dots, n$ , are defined explicitly as in algorithm 6, or according to the algorithm 8. The solution  $x^{(k)} \in \mathbb{R}^n$  of (5.55) lies in the set  $F_X^{(k)}$

$$F \supseteq F_X^{(k)} \quad (5.56)$$

with

$$F_X^{(k)} := F \cap I^{(k)} \quad (5.57)$$

$$I^{(k)} = I_1^{(k)} \times I_2^{(k)} \times \dots \times I_n^{(k)}. \quad (5.58)$$

To assure global convergence of the algorithm, we apply a line search procedure. Where the augmented Lagrangian merit function (5.34) include the feasibility constraints  $e_j$ ,  $j = 1, \dots, m_f$ , it can be defined by several ways. The most popular and communally used functional for the problem (5.53) is given by:

$$\begin{aligned} \Phi_\rho \begin{pmatrix} x \\ y \end{pmatrix} &:= f(x) + \sum_{j=1}^{m_e} \left( (y_c)_j c_j(x) + \frac{(\rho_c)_j}{2} c_j^2(x) \right) \\ &+ \sum_{j=m_e+1}^{m_c} \left\{ \begin{array}{ll} (y_c)_j c_j(x) + \frac{(\rho_c)_j}{2} c_j^2(x), & \text{if } -\frac{(y_c)_j}{(\rho_c)_j} \leq c_j(x) \\ -\frac{(y_c)_j^2}{2(\rho_c)_j}, & \text{otherwise} \end{array} \right. \\ &+ \sum_{j=1}^{m_f} \left\{ \begin{array}{ll} (y_e)_j e_j(x) + \frac{(\rho_e)_j}{2} e_j^2(x), & \text{if } -\frac{(y_e)_j}{(\rho_e)_j} \leq e_j(x) \\ -\frac{(y_e)_j^2}{2(\rho_e)_j}, & \text{otherwise} \end{array} \right. \end{aligned} \quad (5.59)$$

for a given set of penalty parameters

$$\rho := \begin{pmatrix} \rho_c \\ \rho_e \end{pmatrix} \quad (5.60)$$

with  $(\rho_c)_j > 0, j = 1, \dots, m_c$ , and  $(\rho_e)_j > 0, j = 1, \dots, m_f$ . Moreover, we denote the Lagrangian multipliers for the constraints  $c_j(x), j = 1, \dots, m_c$ , and the feasibility constraints  $e_j(x), j = 1, \dots, m_f$  by

$$y := \begin{pmatrix} y_c \\ y_e \end{pmatrix} \quad (5.61)$$

with  $y_c = ((y_c)_1, \dots, (y_c)_{m_c})^T \in \mathbb{R}^{m_c}$ , and  $y_e = ((y_e)_1, \dots, (y_e)_{m_f})^T \in \mathbb{R}^{m_f}$ . The penalty parameters are updated according to Algorithm 10. We can summarize the algorithm adapted to the optimization problems with feasibility constraints as follows:

**Algorithm 12**

- 
- 1: Choose starting point  $x^{(0)} \in \mathbb{R}^n$  and  $y^{(0)}$ . Compute  $f(x^{(0)}), \nabla f(x^{(0)}), c_j(x^{(0)}), \nabla c_j(x^{(0)}), j = 1, \dots, m_c$ . Set parameters  $\omega \in ]0; 1[, r \in (0, 1), \beta \in (0, 1), \tau > 0, c_2 > c_1 > 1$  and penalty parameters  $\rho_j^{(-1)} > 0, j = 1, \dots, m_c$ . Let  $k := 0$
  - 2: Determine  $d^{(k)}$  for the objective and constraints as in Algorithm 6 or by Algorithm 8. Let  $f^{(k)}(x), c_j^{(k)}(x), j = 1, \dots, m_c$  be defined by (5.20) and (5.21). Formulate (5.40) for the corresponding iteration  $k$ .
  - 3: Solve (5.40). Let  $z^{(k)}$  be the optimal solution of (5.40) and  $v^{(k)}$  the vector of the corresponding Lagrangian multipliers.
  - 4: **If**  $z^{(k)} = x^{(k)}$ , then STOP.  $(x^{(k)}, v^{(k)})$  is a KKT point of (5.33).
  - 5: Let  $s_d^{(k)} = \begin{pmatrix} z^{(k)} - x^{(k)} \\ v^{(k)} - y^{(k)} \end{pmatrix}$ ,  $\hat{\delta}^{(k)} := \|z^{(k)} - x^{(k)}\|_2$  and  $\eta^{(k)}$  as defined in (4.6). Let  $i = 0$  and  $\rho^{(k,0)} := \rho^{(k-1)}$ .
  - 6: Compute  $\Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}, \nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}^T s_d^{(k)}$
  - 7: **If**  $\nabla \Phi_{\rho^{(k,i)}} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix}^T s_d^{(k)}$ , update penalty parameters according to Algorithm 10. Let  $i = i + 1$  and go to Step 6. Otherwise, let  $\rho^{(k)} := \rho^{(k,i)}, i = 0$  and  $\sigma^{(k,0)} := 1$
  - 8: Compute  $f(x^{(k)} + \sigma^{(k,i)}(z^{(k)} - x^{(k)})), c_j(x^{(k)} + \sigma^{(k,i)}(z^{(k)} - x^{(k)})), j = 1, \dots, m_c,$   
 $e_j(x^{(k)} + \sigma^{(k,i)}(z^{(k)} - x^{(k)})), j = 1, \dots, m_f$  and  $\Phi_{\rho^{(k)}} \left( \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \sigma^{(k,i)} d^{(k)} \right)$ .
  - 9: **if** (5.42) is not satisfied **then** let  $\sigma^{(k,i+1)} := \beta \sigma^{(k,i)}, i = i + 1$  and repeat (Armijo). Otherwise,  $\sigma^{(k)} := \sigma^{(k,i)}$ .
  - 10: Let  $\begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} := \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \sigma^{(k)} s_d^{(k)}, k := k + 1$ .
  - 11: Compute  $\nabla f(x^{(k)}), \nabla c_j(x^{(k)}), j = 1, \dots, m_c$  and go to Step 1.
-

# CONCLUSION

The main focus of this dissertation is the development of new approach and algorithms based on the method of moving asymptote for solving the unconstrained and constrained optimization problems. The goal of these additional modifications is to improve the computational performance, reduction of calculation time and generalize these approach for a kind of constrained optimization problems, where the feasible region define by an inequality constraints called feasibility constraints. The remaining constraints, so-called regular constraints, may be violated. The classical methods yielding a feasible sequence of iterates. The algorithms proposed generate a sequence of feasible iterates, they require function and gradient evaluations at infeasible points, i.e, at each iteration we check if the iterate point satisfies the feasibility constraints.

The corresponding algorithm solves continuous nonlinear programs iteratively, by a sequence of convex subproblems. On subproblem level, the objective function and the regular constraints are replaced by a rational, convex and separable approximations, and we include the feasibility constraints directly in the subproblems generated by the algorithm. We use a line search procedure to ensure a global convergence with respect to feasibility constraints. Note that the convexity of the subproblems is necessary to have an unique solution and then formulates a new subproblem from this solution.

The corresponding algorithms was adapted to become easy to implement for large scale problems and then can be solved. If exists, the sparse structure of the gradients and the Hessian is exploited. Moreover, linear constraints are approximated optionally. A modification of the approximations and an active set strategy are applied to reduce the size of the subproblems and thus speed up the solution process. To satisfy feasibility constraints in every main iteration, we apply the active strategy only for the regular constraints. A global convergence of the sequence generated is established under basic assumptions, and we show the performance of the algorithms proposed, by some numerical comparisons with a another well-known algorithms.

As future work, is to apply these algorithms in the free materiel optimization (FMO), where sparse, large-scale optimizations problems are to be solved. Proceeding from a finite element discretization the design of a structure is to be optimized, such that it becomes as stiff as possible. The compliance function measures the stiffness of the resulting structure dependent on the material properties in each finite element. In addition, the total amount of material is bounded. Moreover, feasibility constraints are introduced to ensure positive definiteness of the elementary stiffness matrices.





# CONCLUSION "version française"

L'objectif principal de cette thèse est le développement des nouvelles approches et algorithmes basés sur la méthode des asymptotes mobiles pour résoudre les problèmes d'optimisation avec et sans contraintes. Le but de ces modifications est d'améliorer les performances de calcul, de réduire le temps de calcul et de généraliser ces approches pour une sorte de problèmes d'optimisation avec contraintes, où la région réalisable est définie par des contraintes d'inégalité appelée contrainte de faisabilité. Les contraintes restantes, les contraintes dites régulières, peuvent être violées. Les méthodes classiques donnent une séquence d'itérations qui peut ne pas satisfaire ces contraintes de faisabilité. Les algorithmes proposés génèrent une séquence d'itérations réalisables, ils nécessitent des évaluations de fonction et de gradient en des points irréalisables, c'est-à-dire qu'à chaque itération on vérifie si le point d'itération satisfait les contraintes de faisabilité.

L'algorithme correspondant résout les programmes non linéaires continus de façon itérative, par une séquence de sous-problèmes convexes. Au niveau des sous-problèmes, la fonction objective et les contraintes régulières sont remplacées par une approximation rationnelle, convexe et séparable, et nous intégrons les contraintes de faisabilité directement dans les sous-problèmes générés par l'algorithme. Nous utilisons une procédure de recherche de lignes pour assurer une convergence globale par rapport aux contraintes de faisabilité. Notez que la convexité des sous-problèmes est nécessaire pour avoir une solution unique et formule ensuite un nouveau sous-problème à partir de cette solution.

Les algorithmes correspondants ont été adaptés pour devenir faciles à mettre en œuvre pour des problèmes à grande échelle et peuvent ensuite être résolus. S'il existe, la structure clairsemée des gradients et de la Hesse est exploitée. De plus, les contraintes linéaires sont approximées facultativement. Une modification des approximations et une stratégie d'ensemble active sont appliquées pour réduire la taille des sous-problèmes et ainsi accélérer le processus de solution. Pour satisfaire les contraintes de faisabilité dans chaque itération principale, nous appliquons la stratégie active uniquement pour les contraintes régulières. Une convergence globale de la séquence générée est établie à partir d'hypothèses de base, et nous montrons la performance des algorithmes proposés, par quelques comparaisons numériques avec d'autres algorithmes bien connus.

Comme travail futur, il s'agit d'appliquer ces algorithmes dans l'optimisation du matériel libre (FMO), où des problèmes d'optimisations clairsemés et à grande échelle doivent être résolus. En partant d'une discrétisation par éléments finis, la conception d'une structure doit être optimisée, afin qu'elle devienne aussi rigide que possible. La fonction de conformité mesure la rigidité de la structure résultante en fonction des propriétés du matériau dans chaque élément fini. De plus, la quantité totale de matériel est limitée. De plus, des contraintes de faisabilité sont introduites pour assurer une définition positive des matrices élémentaires de rigidité.



# Bibliography

- [1] E. D. Andersen, J. Gondzio, C. Mészáros, X. Xu, et al. *Implementation of interior point methods for large scale linear programming*. HEC/Université de Genève, 1996.
- [2] L. Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific Journal of mathematics*, 16(1):1–3, 1966.
- [3] M. Bachar, T. Estebenet, and A. Guessab. A moving asymptotes algorithm using new local convex approximation methods with explicit solutions. *Electronic Transactions on Numerical Analysis*, 43:21–44, 2014.
- [4] S. Bakhtiari and A. Tits. A simple primal-dual feasible interior-point method for nonlinear programming with monotone descent. *Computational Optimization and Applications*, 25(1-3):17–38, 2003.
- [5] R. Bartlett, A. Wachter, and L. Biegler. Active set vs. interior point strategies for model predictive control. In *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No. 00CH36334)*, volume 6, pages 4229–4233. IEEE, 2000.
- [6] M. P. Bendsoe, J. Guedes, R. B. Haber, P. Pedersen, and J. Taylor. An analytical model to predict optimal material properties in the context of optimal structural design. 1994.
- [7] D. P. Bertsekas and A. Scientific. *Convex optimization algorithms*. Athena Scientific Belmont, 2015.
- [8] E. G. Birgin and J. M. Martínez. Large-scale active-set box-constrained optimization method with spectral projected gradients. *Computational Optimization and Applications*, 23(1):101–125, 2002.
- [9] E. G. Birgin, J. M. Martínez, and M. Raydan. Nonmonotone spectral projected gradient methods on convex sets. *SIAM Journal on Optimization*, 10(4):1196–1211, 2000.
- [10] K.-U. Bletzinger. Extended method of moving asymptotes based on second-order information. *Structural optimization*, 5(3):175–183, 1993.
- [11] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, USA, 2004.
- [12] T. E. Bruns and D. A. Tortorelli. Topology optimization of non-linear elastic structures and compliant mechanisms. *Computer methods in applied mechanics and engineering*, 190(26-27):3443–3459, 2001.
- [13] M. Bruyneel and P. Duysinx. Note on topology optimization of continuum structures including self-weight. *Structural and Multidisciplinary Optimization*, 29(4):245–256, 2005.
- [14] M. Bruyneel, P. Duysinx, and C. Fleury. A family of mma approximations for structural optimization. *Structural and Multidisciplinary Optimization*, 24(4):263–276, Oct 2002.
- [15] R. Burden and J. Faires. Numerical analysis (cengage learning, inc, boston). *Google Scholar*, 2011.
- [16] C. Chen, M. Li, X. Liu, and Y. Ye. Extended admm and bcd for nonseparable convex minimization models with quadratic coupling terms: convergence analysis and insights. *Mathematical Programming*, pages 1–41, 2015.
- [17] H. Chickermane and H. Gea. Structural optimization using a new local approximation method. *International journal for numerical methods in engineering*, 39(5):829–846, 1996.

- [18] S. Ertel, K. Schittkowski, and C. Zillober. Sequential convex programming methods for free material optimization. 2007.
- [19] R. Fletcher. An exact penalty function for nonlinear programming with inequalities. *Mathematical Programming*, 5(1):129–150, 1973.
- [20] R. Fletcher. An ideal penalty function for constrained optimization. *IMA Journal of Applied Mathematics*, 15(3):319–342, 1975.
- [21] R. Fletcher and S. Leyffer. Nonlinear programming without a penalty function. *Mathematical programming*, 91(2):239–269, 2002.
- [22] R. Fletcher, S. Leyffer, and P. L. Toint. On the global convergence of a filter–sqp algorithm. *SIAM Journal on Optimization*, 13(1):44–59, 2002.
- [23] C. Fleury. Shape optimal design by the convex linearization method. In *The Optimum Shape*, pages 297–326. Springer, 1986.
- [24] C. Fleury. Conlin: an efficient dual optimizer based on convex approximation concepts. *Structural optimization*, 1(2):81–89, 1989.
- [25] C. Fleury. Efficient approximation concepts using second order information. *International Journal for Numerical Methods in Engineering*, 28(9):2041–2058, 1989.
- [26] C. Fleury. First and second order convex approximation strategies in structural optimization. *Structural optimization*, 1(1):3–10, 1989.
- [27] C. Fleury. Mathematical programming methods for constrained optimization: dual methods. *Structural optimization: status and promise, progress in Astronautics and Aeronautics*, 1993.
- [28] C. Fleury. Structural optimization methods for large scale problems: status and limitations. In *ASME 2007 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages 513–522. American Society of Mechanical Engineers, 2007.
- [29] C. Fleury and V. Braibant. Structural optimization: a new dual method using mixed variables. *International journal for numerical methods in engineering*, 23(3):409–428, 1986.
- [30] P. E. Gill and E. Wong. Sequential quadratic programming methods. In *Mixed integer nonlinear programming*, pages 147–224. Springer, 2012.
- [31] M. A. Gomes-Ruggiero, M. Sachine, and S. A. Santos. Globally convergent modifications to the method of moving asymptotes and the solution of the subproblems using trust regions: theoretical and numerical results. *Revised Version December*, 3, 2010.
- [32] M. A. Gomes-Ruggiero, M. Sachine, and S. A. Santos. A spectral updating for the method of moving asymptotes. *Optimization Methods and Software*, 25(6):883–893, 2010.
- [33] J. Gondzio. Interior point methods 25 years later. *European Journal of Operational Research*, 218(3):587–601, 2012.
- [34] A. Greenbaum and T. P. Chartier. *Numerical methods: design, analysis, and computer implementation of algorithms*. Princeton University Press, 2012.
- [35] L. Grippo and S. Lucidi. A globally convergent version of the polak-ribiere conjugate gradient method. *Mathematical Programming*, 78(3):375–391, 1997.
- [36] A. Guessab and A. Driouch. A globally convergent modified multivariate version of mma. *Applicable Mathematics and Discrete Mathematics*, 2020.
- [37] A. Guessab, A. Driouch, and O. Nouisser. A globally convergent modified version of the method of moving asymptotes. *Applicable Analysis and Discrete Mathematics*, (00):42–42, 2019.

- [38] J. Herskovits. A two-stage feasible directions algorithm for nonlinear constrained optimization. *Mathematical Programming*, 36(1):19, 1986.
- [39] J. Herskovits. Feasible direction interior-point technique for nonlinear optimization. *Journal of optimization theory and applications*, 99(1):121–146, 1998.
- [40] M. Jamil and X.-S. Yang. A literature survey of benchmark functions for global optimisation problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, 4(2):150–194, 2013.
- [41] J. Jinbao, Z. Kecun, and X. Shengjia. A superlinearly and quadratically convergent sqp type feasible method for constrained optimization. *Applied Mathematics-A Journal of Chinese Universities*, 15(3):319–331, 2000.
- [42] C. Kiyono, S. Vatanabe, E. Silva, and J. Reddy. A new multi-p-norm formulation approach for stress-based topology optimization design. *Composite Structures*, 156:10 – 19, 2016. 70th Anniversary of Professor J. N. Reddy.
- [43] M. Kočvara and M. Stingl. Solving stress constrained problems in topology and material optimization. *Structural and Multidisciplinary Optimization*, 46(1):1–15, 2012.
- [44] M. Kojima, S. Mizuno, and A. Yoshise. An iteration potential reduction algorithm for linear complementarity problems. *Mathematical programming*, 50(1-3):331–342, 1991.
- [45] A. N. Laboratory, U. S. D. of Energy. Office of Scientific, and T. Information. *DRVOCR: A FORTRAN Implementation of Davidon's Optimally Conditioned Method. [Minimizes Function of N Variables Using Function Values and Gradients]*. Argonne National Laboratory, 1977.
- [46] S. Lehmann. *A strictly feasible sequential convex programming method*. PhD thesis, 2011.
- [47] Z. Ling, X. Ronglu, W. Yi, and A. El-Sabbagh. Topology optimization of constrained layer damping on plates using method of moving asymptote (mma) approach. *Shock and Vibration*, 18(1-2):221–244, 2011.
- [48] Q. Ni. A globally convergent method of moving asymptotes with trust region technique. *Optimization methods and software*, 18(3):283–297, 2003.
- [49] Q. Ni, C. Zillober, and K. Schittkowski. Sequential convex programming methods for solving large topology optimization problems: implementation and computational results. *Journal of Computational Mathematics*, pages 491–502, 2005.
- [50] E. R. Panier and A. L. Tits. On combining feasibility, descent and superlinear convergence in inequality constrained optimization. *Mathematical programming*, 59(1-3):261–276, 1993.
- [51] E. R. Panier, A. L. Tits, and J. N. Herskovits. A qp-free, globally convergent, locally superlinearly convergent algorithm for inequality constrained optimization. *SIAM Journal on Control and Optimization*, 26(4):788–811, 1988.
- [52] M. Papadrakakis, N. D. Lagaros, Y. Tsompanakis, and V. Plevris. Large scale structural optimization: computational methods and optimization algorithms. *Archives of Computational Methods in Engineering*, 8(3):239–301, 2001.
- [53] I. Pólik and T. Terlaky. Interior point methods for nonlinear optimization. In *Nonlinear optimization*, pages 215–276. Springer, 2010.
- [54] F. A. Potra and S. J. Wright. Interior-point methods. *Journal of Computational and Applied Mathematics*, 124(1-2):281–302, 2000.
- [55] L. M. Rios and N. V. Sahinidis. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3):1247–1293, 2013.
- [56] S. Rojas-Labanda and M. Stolpe. Benchmarking optimization solvers for structural topology optimization. *Structural and Multidisciplinary Optimization*, 52(3):527–547, 2015.

- [57] J. Rosen. The gradient projection method for nonlinear programming. part ii. nonlinear constraints. *Journal of the Society for Industrial and Applied Mathematics*, 9(4):514–532, 1961.
- [58] J. B. Rosen. The gradient projection method for nonlinear programming. part i. linear constraints. *Journal of the society for industrial and applied mathematics*, 8(1):181–217, 1960.
- [59] R. Salomon. Re-evaluating genetic algorithm performance under coordinate rotation of benchmark functions. a survey of some theoretical and practical aspects of genetic algorithms. *BioSystems*, 39(3):263–278, 1996.
- [60] K. Schittkowski. On the convergence of a sequential quadratic programming method with an augmented lagrangian line search function. *Mathematische Operationsforschung und Statistik. Series Optimization*, 14(2):197–216, 1983.
- [61] K. Schittkowski and C. Zillober. Nonlinear programming: algorithms, software, and applications. In *IFIP Conference on System Modeling and Optimization*, pages 73–107. Springer, 2003.
- [62] K. Schittkowski, C. Zillober, and R. Zotemantel. Numerical comparison of nonlinear programming algorithms for structural optimization. *Structural Optimization*, 7(1-2):1–19, 1994.
- [63] O. Sigmund. A 99 line topology optimization code written in matlab. *Structural and multidisciplinary optimization*, 21(2):120–127, 2001.
- [64] O. Sigmund. Topology optimization—theory, methods, and applications, 2003.
- [65] O. Sigmund and K. Maute. Topology optimization approaches. *Structural and Multidisciplinary Optimization*, 48(6):1031–1055, 2013.
- [66] H. Smaoui, C. Fleury, and L. Schmit. Advances in dual algorithms and convex approximation methods. In *Advanced Marine Systems Conference*, page 2382, 1988.
- [67] M. Stingl. A generalized augmented lagrangian method for semidefinite programming.
- [68] M. Stingl, M. Kočvara, and G. Leugering. A sequential convex semidefinite programming algorithm with an application to multiple-load free material optimization. *SIAM Journal on Optimization*, 20(1):130–155, 2009.
- [69] K. Svanberg. The method of moving asymptotes—a new method for structural optimization. *International journal for numerical methods in engineering*, 24(2):359–373, 1987.
- [70] K. Svanberg. A globally convergent version of mma without linesearch. In *Proceedings of the first world congress of structural and multidisciplinary optimization*, volume 28, pages 9–16. Goslar, Germany, 1995.
- [71] K. Svanberg. The method of moving asymptotes-modelling aspects and solution schemes. *Lecture Notes for the DCAMM course Advanced Topics in Structural Optimization*, 1998.
- [72] K. Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM journal on optimization*, 12(2):555–573, 2002.
- [73] K. Svanberg. Mma and gmma, versions september 2007. *Optimization and Systems Theory*, 104, 2007.
- [74] D. M. Topkis and A. F. Veinott, Jr. On the convergence of some feasible direction algorithms for nonlinear programming. *SIAM Journal on Control*, 5(2):268–279, 1967.
- [75] G. N. Vanderplaats and F. Moses. Structural optimization by methods of feasible directions. *Computers & Structures*, 3(4):739–755, 1973.
- [76] H. Wang and Q. Ni. A new method of moving asymptotes for large-scale unconstrained optimization. *Applied Mathematics and Computation*, 203(1):62–71, 2008.
- [77] D. Whitley, S. Rana, J. Dzuber, and K. E. Mathias. Evaluating evolutionary algorithms. *Artificial intelligence*, 85(1-2):245–276, 1996.

- 
- [78] P. H. Winston. *Artificial Intelligence, 3rd ed.* Addison-Wesley, 1992.
- [79] X. Yao and Y. Liu. Fast evolution strategies. In *International Conference on Evolutionary Programming*, pages 149–161. Springer, 1997.
- [80] W.-H. Zhang and C. Fleury. A modification of convex approximation methods for structural optimization. *Computers & Structures*, 64(1-4):89–95, 1997.
- [81] Z. Zhu. An efficient sequential quadratic programming algorithm for nonlinear programming. *Journal of Computational and Applied Mathematics*, 175(2):447–464, 2005.
- [82] C. Zillober. A globally convergent version of the method of moving asymptotes. *Structural optimization*, 6(3):166–174, 1993.
- [83] C. Zillober. A combined convex approximation—interior point approach for large scale nonlinear programming. *Optimization and Engineering*, 2(1):51–73, 2001.
- [84] C. Zillober. Global convergence of a nonlinear programming method using convex approximations. *Numerical Algorithms*, 27(3):265–289, 2001.
- [85] C. Zillober. Scpip—an efficient software tool for the solution of structural optimization problems. *Structural and Multidisciplinary Optimization*, 24(5):362–371, 2002.
- [86] C. Zillober, K. Schittkowski, and K. Moritzen. Very large scale optimization by sequential convex programming. *Optimization Methods and Software*, 19(1):103–120, 2004.
- [87] G. Zoutendijk. *Methods of feasible directions: a study in linear and non-linear programming*. Elsevier, 1960.
- [88] J. Zowe, M. Kočvara, and M. P. Bendsøe. Free material optimization via mathematical programming. *Mathematical programming*, 79(1-3):445–466, 1997.



