Nelder-Mead Optimization under Linear Constraints

Ebert Brea

September 2004

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

FACULTY OF ENGINEERING, SCIENCE AND MATHEMATICS

SCHOOL OF MATHEMATICS

Doctor of Philosophy

NELDER-MEAD OPTIMIZATION UNDER LINEAR CONSTRAINTS

by Ebert Brea

An extension of the Nelder-Mead simplex algorithm is presented in this dissertation. The algorithm was developed for minimizing a non-linear objective function subject to linear inequality constraints. The algorithm assumes that the objective function is analytically unavailable or its evaluation at each experimental design point is very expensive. The algorithm generates a feasible trial point at each iteration and compares it with the current pattern of points (simplex). The algorithm, called the Linear Constraint Nelder-Mead (LCNM), takes advantage of when the current simplex is eventually flattened by a constrained reflection or expansion operation, as a result of meeting the boundary of the feasible region. When this occurs, the algorithm reduces the number of vertices of the current simplex thereby avoiding the degeneration of the simplex. A limited study of its performance is developed by case studies. Although the LCNM algorithm was designed for minimizing linearly constrained non-linear objective functions, a particular case of linear programming is theoretically studied, showing that a very slow convergence rate is possible. A modification to the LCNM algorithm was included for improving the convergence rate of the algorithm. Two variations of the algorithm were also investigated. The LCNM algorithm has displayed a good enough performance when the objective function is corrupted by noise. This fact allows us to appreciate the LCNM algorithm as an optimization method for problems of optimization by simulation, where the evaluation of the design points can require a huge computational effort.

To Marlene and my parents

Contents

List of Tables

List of Figures

Acknowledgements

I wish to thank Professor Russell C. H. Cheng, my supervisor, for his invaluable comments, suggestions, time, continuous help and guidance during my study and research at the School of Mathematics of the University of Southampton.

I want to thank Dr Owen Jones, my adviser, for his opportune help during my work at the University of Southampton.

I also wish to express my thankful words to all the staff of the School of Mathematics of the University of Southampton for their assistance through my permanence at the University of Southampton.

I also want to thank to Marlene, my wife, who has always been with me, giving her words of help.

I cannot miss out my family and Marlene's family who have always been giving me their help during my lovely stay in Southampton. For all them, my profoundest words of gratitude.

The author is supported by the Council for Scientific and Humanistic Development of the Central University of Venezuela.

Figure 3.2: General flow chart of the LCNM algorithm on page 43 has been reproduced with permission from the United Kingdom Simulation Society.

List of Acronyms

BBD Box-Behnken Design

DDSE descent direction of the simplex edges

DEDS discrete event dynamic system

DFRS distribution-free rank sum

DTP distance to the true point

GOFH gradient of the objective function hyperplane

KKT Karush-Kuhn-Tucker

KT Kuhn-Tucker

LCM linearly constrained minimization

LCNM Linear Constrained Nelder-Mead

LCNM-G Linear Constrained Nelder-Mead method without using the building of a simplex in the intersection space

LCNM-R Linear Constrained Nelder-Mead method without reducing simplex vertices when it has activated any new constraint

LCNM-RG Linear Constrained Nelder-Mead method without building of simplex in the intersection space and reducing of vertices

LCNM+PC Linear Constrained Nelder-Mead method with Premature Collapse

LCP Linear Constraint Procedure

LPP Linear Programming Problem

LST Lucidi-Sciandrone-Tseng

NE number of function evaluations

NM Nelder-Mead

NP Nested Partitions

PCP Premature Collapse Procedure

PM Performance Measure

RFIC repeated focused inside contraction

RSM response surface methodology

SDES Stochastic Discrete Event Simulation

SM Subrahmanyam method

TP true point

Chapter 1

Introduction

The identification of the optimum operation of a complex system often requires the application of algorithms able to search for the best amongst a very extensive set of feasible options. This process of identifying the best setting of a system is commonly named optimization.

Herein, the development of an optimization method will be considered for a non-linear function subject to linear inequality constraints. It is assumed that the analytical expression of the objective function is unavailable or that its evaluation at each experimental design point is very expensive, such as occurs in some simulation models of complex stochastic systems.

Moreover, such simulation models produce an objective function corrupted by noise, where the accuracy of the estimation of the objective function depends on the number of replications at each design point.

Methods of searching for the optimal performance in complex systems must consider two important aspects: its accuracy and cost. This latter is measured most easily by the number of performance measure (objective function) evaluations we have to make in the search for the optimum. A single evaluation can already be expensive if it requires a simulation run of a complex model.

Hence, the aims of our research are:

- To investigate optimization algorithms appropriate for our requirements.
- To develop an algorithm that needs relatively few function evaluations and which can handle linearly constrained non-linear problems.

- To study the behaviour of the algorithm.
- To develop if necessary modifications of the algorithm.
- To establish the performance of the algorithm when the objective function is affected by noise.

These aims induce us to first examine a set of heuristic methods of optimization with the aim of identifying their properties, and so to guide our research.

As a result of this study, we consider appropriate for our purpose, the development of a constrained optimization algorithm based on the widely known Nelder-Mead (NM) simplex method (Nelder and Mead 1965), because of its economy in function evaluations per iteration and its good performance when the objective function is in the presence of noise (Humphrey and Wilson 2000).

The main contribution of this research is the development of an optimization algorithm called Linear Constrained Nelder-Mead (LCNM) algorithm, which was designed for identifying the optimum solution to non-linear objective function subject to linear inequality constraints. The LCNM algorithm has also shown a good performance in the identification of the optimum when the objective function is affected by noise.

The LCNM algorithm takes advantage of the eventual collapse of the simplex onto the boundary of the feasible region as a consequence of having reached the boundary. It does so by reducing the number of vertices of the simplex when a collapse occurs. It identifies the linear constraints that have been reached by the vertices of the simplex and then explores the sub-space defined by these reached (activated) linear constraints. This action minimizes the possibility of degenerating the simplex, because its number of vertices are adapted to the dimension of sub-space defined by the activated constraints.

To design the LCNM algorithm, a theoretical framework was developed for defining the new algorithm and comparing it with other optimization algorithms.

Furthermore, variations of the algorithm were explored.

The Nelder-Mead method can of course be applied to non-smooth objective function, but this has not been the form of the present thesis. Instead the LCNM algorithm was tested by a large number of numerical experiments using smooth monomodal and smooth multimodal objective functions. This class of situations can emerge from real and practical problems, such as complex and stochastic systems represented by simulation models, where the response surface of the performance of the models is often smooth but is affected by noise.

The remaining chapters are as follows. Chapter 2 gives a review of some of the advantages and disadvantages of the so-called free-derivatives methods and line search methods. The reasons that induced us to consider the NM simplex method are given in this chapter. A statement of the problem, a description of the LCNM algorithm and some preliminary numerical examples are presented in Chapter 3. Some general properties of the algorithm and a study of its behaviour through case studies are developed in Chapter 4. In particular, a theoretical study when the LCNM algorithm is applied to a linear constrained linear minimization problem has demonstrated an unexpectedly slow convergence to its minimum. Chapter 5 deals with a modified version of the LCNM algorithm, which is founded on the induction of the collapse of the simplex prematurely for improving the convergence rate of the algorithm. In addition, a number of test problems were carried out for comparing the LCNM algorithm and its premature collapse version. Chapter 6 concerns two further variations of the LCNM algorithm, whose underlying principles are shown, and results of numerical test problems are reported comparing them with the LCNM algorithm. The performance of our algorithm when the objective function is affected by noise is described in Chapter 7, through comparing the LCNM algorithm and its premature collapse version. Finally, conclusions and future research are given in Chapter 8.

Chapter 2

A review of some approaches

2.1 Introduction

The development of new methods for finding answer to optimization problems could begin with a taxonomic study and analysis of the different approaches for identifying their own essence. We shall throughout the dissertation consider function minimization.

Lewis et al. (2000) classify the methods of optimization according to the Taylor expansion of the objective function employed by the method. From this perspective, the method of Newton can be classified as a second-order method, because it employs the second-order Taylor series for identifying the descent direction of the objective function (Nocedal and Wright 1999). The method of Davidon-Fletcher-Power, the conjugate gradient method, the method of Zoutendijk, among others (Bazaraa and Shetty 1979), are clear examples of first-order methods. The so-called random search methods (Boender and Romeijn 1995), the Nested Partitions (NP) method (Shi and Ólafsson 2000a), the pattern search methods (Audet and Dennis Jr 2003), the Nelder-Mead (NM) simplex method (Nelder and Mead 1965), the response surface methodology (RSM) (Myers and Montgomery 2002) and the objective-derivative-free method of Lucidi et al. (2002) constitute some examples of methods of zero-order, because they do not use the expansion of series of Taylor of the objective function for identifying the descent direction of the objective function. However, the sequential procedure RSM of Box and Wilson (1951) could be in the group of first-order methods, because it employs first-order Taylor series of the objective function for moving to a near stationary region.

This taxonomic framework would allow us to recognize the essence of an optimization method for knowing their requirements regarding of the number of function evaluations per iteration. This constitutes an important factor for choosing an optimization approach, especially when the objective function is analytically unavailable or it is expensive to obtain the value of the function at any point.

In this dissertation we shall study only the called methods of zero-order. These provide a way for obtaining the optimum without using the objective function gradient. These methods can be classified into two large group of methods: "direct search methods" and "derivative-free methods". This latter group can be distinguished from the direct search methods, due to fact that the direct search methods optimize the objective function by evaluation of trial points, whilst the called derivative-free methods find it by "line search", which can be defined as the procedure for minimizing a function along a direction (Bazaraa and Shetty 1979).

The layout of this chapter is as follows. In Section 2.2 some zero-order methods, such as, the NP method, the pattern search methods, the NM simplex method and the Lucidi-Sciandrone-Tseng (LST) method are briefly introduced. However, the NP method is presented in more detail due to the fact that we were initially interested in it as a feasible method for solving constrained optimization problems. A short explanation of the RSM is presented in Section 2.3. Finally, the reason for why we decided to concentrate on the NM method for constrained optimization are considered in Section 2.4.

2.2 The methods of zero-order

The methods of zero-order have had an important role in the optimization of systems, especially when evaluation of the objective function is computational expensive or the analytical expression of the objective function is unavailable.

They are occasionally called derivative-free methods or direct search methods. This latter term, introduced by Hooke and Jeeves (1961), can be defined as those methods based on the evaluation of trial points for comparing them with the current value. These trial values are used to establish the direction of search of the next step.

Hooke and Jeeves (1961) pointed out five reasons why they studied the methods of direct search, and these are summarized herein:

- Direct search methods are most useful for the case when calculation of derivatives is costly or impossible, or simply there is not an explicit expression of the objective function.

- They can be easily employed in computers.
- They can supply an approximate solution, which can be improved at each stage of the method.

In addition, Hooke and Jeeves (1961) have included a formal definition of direct search, whose reading we consider important, due to its validity to date.

Trosset (1997) proposes a short definition of direct search, which can be summarized as the methods of optimization that are mainly determined by ranking of the objective function. Furthermore, he points out some ambiguities on the definitions of direct search that are usually presented in the literature. However, the author remarks on a definition of M. H. Wright, who defines the direct search methods as the set of methods that do not calculate any approximation of the gradient.

This standpoint excludes those quasi-Newton methods based on finite-difference, which are occasionally considered direct search methods. However, in this work, the so-called finite-difference quasi-Newton methods will be considered non-direct search methods.

In (Lewis et al. 2000), the authors indicate three reasons why the methods of direct search are still popularly employed by practitioners, namely:

- Direct search methods work well in practical problems and many of them are demonstrated to have global convergence.
- Methods based on a quasi-Newton approach can be applied to some varieties of non-linear optimization problems.
- Direct search methods are moderately easy to implement. They do not need very complex algorithms for calculating derivatives, as in quasi-Newton methods.

A novel perspective on the study of the methods of direct search is introduced by Kolda et al. (2003), who present the named *generating set search* for developing a general and flexible theory of direct search methods, and which allows us to analyse some pattern search methods, such as, generalized pattern search methods (Torczon 1997), evolutionary operation (Box 1957), the pattern search method of Hooke and Jeeves (1961), among others.

In addition characteristics that can be considered in the comparative study of direct search methods are: the implementation of the methods, its development for constrained optimization problems, and the number of function evaluations.

2.2.1 The Nested Partitions method

According to Boender and Romeijn (1995), random search methods can be defined as an algorithmic method that optimizes a function through points whose generation have been previously specified by some probability distribution within a feasible region.

We can interpret this class of methods as founded on the generation of trial points from a probability distribution, with points constituting an independently and identically distributed sequence. Examples of this class of methods are: pure random search, random search, pure adaptive search, adaptive search and the family of simulated annealing methods (Boender and Romeijn 1995). Another family of random search algorithms, called Monte Carlo methods, are widely explained in Rubinstein (1992).

In particular, the Nested Partitions (NP) method (Shi and Ólafsson 2000a) is a type of random search method. This method has been widely studied by Brea (2002) during the course of this investigation. For this reason, we have considered convenient to present it in detail.

The NP method proposed by Shi and Ólafsson (2000a) was first developed for solving global optimization problems in deterministic models. A modification of the method for solving problems concerning the optimal allocation of resources in discrete event dynamic system (DEDS) is discussed by Shi and Chen (2000). The method systematically partitions the feasible region and concentrates the search for an optimal solution in regions that are most promising. Examination of the most promising region is done through random sampling in the considered feasible region. Furthermore, the NP method offers convergence with probability one in finite time, without verifying all the feasible regions (Shi and Ólafsson 2000b).

The NP algorithm has four main steps at each kth iteration. These steps are: partitioning, random sampling, ranking and selection of the best design point and, further partition, backtracking or stopping. A practical example of the NP method is shown in the problem developed by Brea and Cheng (2003b), who apply the NP method for identifying the optimal operation of a four-queue network.

a) Partitioning

The first step divides the current most promising region $\sigma(k)$ into M_k subregions $\sigma_i(k)$, $i = 1, \ldots, M_k$, and aggregates the surrounding region or complement subset of $\sigma(k)$, denoted by $\Theta \backslash \sigma(k)$, into one. Therefore, within each kth iteration, we have $M_k + 1$ disjoint subsets that define the feasible region denoted as Θ. The method of partitioning employed in this step is very important, because the rate of convergence of the algorithm will depend on the manner of how the different design points of the problem are grouped. These grouping of feasible solutions, according to the performance of the system, can help the algorithm to concentrate the search into the most promising subregion more quickly.

b) Random sampling

The second step of the algorithm randomly chooses design points from each subregion $\sigma_i(k)$, $i = 1, \ldots, M_k$ and from the surrounding region $\sigma_{M_k+1}(k) = \Theta \backslash \sigma(k)$. In this step the NP algorithm is flexible enough, due to the fact that we can employ any scheme of sampling, from our knowledge of the system or from any heuristic. A discussion of a weighted sampling scheme is presented by Shi and Chen (2000).

c) Ranking and selection of the best design

The third step is to choose the most promising region, among the subregions $\sigma_i(k)$, $i = 1, \ldots, M_k$ and the surrounding region $\sigma_{M_k+1}(k)$, by comparing the performance measure (objective function) at each chosen design point.

The method estimates an index $i = 1, \ldots, M_k + 1$ of the most promising either subregion or surrounding region. Here the NP algorithm offers flexibility in the method to be employed in order to determine such an index. A way for finding this index is through statistical comparison of the performance measure for all the sampled design points. Therefore, the method estimates whether the best design point comes from a given subregion or surrounding region. An improvement of the NP method was proposed by Shi and Chen (2000) based on the optimal computing budget allocation (Chen et al. 2000).

d) Further partition, backtracking or stopping

The fourth step of the method is to do further partitions of the subregion $\sigma_B(k)$, where the index B means the index of the most promising subregion, for concentrating the search into the selected subregion. If the best design point belongs to the surrounding region, the method backtracks to a large region containing $\sigma_B(k)$, which is called the superregion of $\sigma(k)$. Two backtracking rules are suggested in (Shi and Chen 2000). In addition, there exists a stopping rule proposed by Shi and Ólafsson (2000b), which is based on the Markov Chain Monte Carlo approach. This shows that the NP method generates a Markov Chain when there exists an unique optimum to the problem.

Implementation of the Nested Partitions algorithm

A limited study of the performance of the NP algorithm was carried out for optimization problems where the decision variables can be discrete or continuous. To do this we developed an algorithm using Visual C++ (Sphar 1999). It was convenient to do this because the Visual C++ has an environment for programming interfaces with the user.

The implemented algorithm is a variant of the original NP algorithm, because we required a software that allows the optimization to problems of both discrete decision variables and continuous decision variables. In the case of continuous decision variables, the method alternately divides each variable range into a pre-specific number of subregions $N_{subrange}$, making approximately the same number of partitions for each variable range. With respect to the sampling of design points, a uniform sampling scheme was used.

Without loss of generality, we shall describe the method through an example of two continuous decision variables. Suppose we have two continuous decision variables θ_1 and θ_2 which can take values between L_i and U_i for $i = 1, 2$, and we choose $N_{subrange} = 2$. In this case, the original feasible region is

$$\sigma(0) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq U_1,\ L_2 \leq \theta_2 \leq U_2\}$$

For the considered region, we assume two subregions $\sigma_1(0)$ and $\sigma_2(0)$ expressed by

$$\begin{array}{l} \sigma_1(0) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1,\ L_2 \leq \theta_2 \leq U_2\} \\ \sigma_2(0) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid m_1 < \theta_1 \leq U_1,\ L_2 \leq \theta_2 \leq U_2\} \end{array},$$

where m_1 is the mean point between L_1 and U_1.

Suppose that after one iteration, the most promising subregion was $\sigma_1(0)$. This is then partitioned into two subregions in the next iteration, but in the direction of θ_2, that is:

$$\begin{array}{l} \sigma_1(1) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1,\ L_2 \leq \theta_2 \leq m_2\} \\ \sigma_2(1) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid L_1 \leq \theta_1 \leq m_1,\ m_2 < \theta_2 \leq U_2\} \end{array},$$

where m_2 is the mean point between L_2 and U_2. The superregion is given by

$$\sigma_3(1) = \{\boldsymbol{\theta} \in \mathbb{R}^2 \mid m_1 < \theta_1 \leq U_1,\ L_2 \leq \theta_2 \leq U_2\}$$

This process is repeated alternately for θ_1 and θ_2 until the size of the side of the largest promising subregion satisfies a value of tolerance fixed by the user. If the algorithm chooses the superregion as the most promising region, it backtracks to the entire feasible region.

In the case that there exist continuous and discrete variables, the method works in the same manner according to the NP algorithm.

Numerical example

To test the implemented algorithm, we carried out two sets of experiments. The first considered the case where the decision variables are discrete whilst the second considered continuous decision variables. We studied the following problem

$$\begin{aligned} &\min_\theta \theta^T Q\theta, \\ \text{subject to:} \quad &\theta \in [-10, 10]^n, \end{aligned}$$

where the matrix Q is a diagonal matrix whose entries $q_{ii} = i \ \forall i = 1, \ldots, n$, that is, $diag(1, \ldots, n)$.

Note that the global minimum point is at the origin.

The number of sampling per subregion (Sampling/r) and the number of sampling in the superregion (Sampling/R) were fixed for each case study.

a) Case: Discrete decision variables

Table 2.1: Summary of experimentation: discrete decision variables

Dimension n	2	2	3	3	3	3	4	4	4	4
Sampling/r	5	10	5	5	10	10	5	5	10	10
Sampling/R	5	5	2	2	2	2	2	2	2	2
Sampling	126	231	1165	5359	1772	13290	1631	2922	5079	11514
Backtracking	0	0	4	22	3	29	4	9	7	17
θ_1	0	0	-2	1	1	0	-3	-1	1	-1
θ_2	0	0	-2	0	0	1	-2	0	-1	0
θ_3			0	0	0	0	0	0	0	0
θ_4							0	0	0	0

Table 2.1 gives a summary of the results obtained for this experimentation, including the total number of function evaluations (Sampling), and the number of times that the

algorithm backtracked (Backtracking). Note that the maximum number of design experiment points is equal to 21^n, and its convergence was close to its global minimum in most cases.

b) Case: Continuous decision variables

In this case, a maximum size of subregion was fixed equal to 0.01 for carrying out the experiments.

Table 2.2: Summary of experimentation: continuous decision variables

Dimension n	2	2	3	3	4	4	5	5
Sampling/r	50	50	50	50	50	50	50	50
Sampling/R	50	50	50	50	50	50	50	50
Sampling	2800	2800	5409	7212	54723	52518	255499	519809
Backtracking	0	0	2	4	39	35	147	310
θ_1	0.09	0.11	0.26	-0.08	0.76	0.06	0.36	-3.13
θ_2	0.00	0.00	0.06	-0.02	1.73	0.99	0.97	-1.65
θ_3			0.00	0.00	-0.13	-0.05	-0.59	-0.36
θ_4					0.00	0.00	-0.01	0.22
θ_5							0.00	0.00

Table 2.2 shows a summary of the yielded results by the algorithm for different scenarios of dimension n and two replications by scenario. It can be seen that a large number of function evaluations was required for each case.

As a result of numerical examples studied by Brea (2002), we would tend to the view that the performance of the NP algorithm is not good enough. The algorithm needs to do a significant number of evaluations of the objective function, and is expensive in terms of function evaluations. Hence we consider, the NP algorithm is not appropriated when it is applied to optimization problems of continuous decision variables.

Moreover, in the case of continuous decision variables, the NP algorithm evaluates $S + s \cdot N_{subrange}$ times the objective function at each iteration, where S is the number of sampling of the superregion, s is the number of sampling per subregion and $N_{subrange}$ is the pre-specific number of partitions for each variable range.

This fact is evidenced by Shi and Ólafsson (2000a), where the authors report results of a constrained optimization problem of the so-called function of Goldstein-Price, whose analytical expression is given by

$$f(\theta_1,\theta_2) = \left[1+(\theta_1+\theta_2+1)^2(19-14\theta_1+3\theta_1^2-14\theta_2+6\theta_1\theta_2+3\theta_2^2)\right]\cdot \left[30+(2\theta_1-3\theta_2)^2(18-32\theta_1+12\theta_1^2+48\theta_2-36\theta_1\theta_2+27\theta_2^2)\right] \quad \forall\theta\in\mathbb{R}^2 \tag{2.1}$$

The constrained minimizing problem considered by Shi and Ólafsson (2000a) is the following

$$\begin{aligned} &\min_\theta f(\theta_1,\theta_2),\\ \text{subject to: } &-2.5\le\theta_i\le 2.5 \quad \forall i=1,2, \end{aligned}$$

where $f(\theta_1,\theta_2)$ is given by Equation (2.1).

According to numerical results reported by Shi and Ólafsson (2000a), the NP algorithm required between 617 and 68900 average function evaluations in 100 replications, for a pre-specific number of subregions equal to four, a specified sampling per region of 25, 35 and 45, and a predetermined tolerance of 0.1, 0.01 and 0.001.

However, the NP algorithm might be employed as an initial tool for the search of optima, in which the objective function has several local optimum points, because the NP algorithm is able to identify the promising region that contains the global optimum point.

2.2.2 Pattern search methods

Pattern search methods were originally developed for solving unconstrained non-linear optimization problems using factorial designs (Box 1957). However, a recent analysis of convergence, founded on a generalized pattern search, was presented by Torczon (1997). Its application to bound constrained optimization problems was developed by Lewis and Torczon (1999) and to linearly constrained optimization problems by Lewis and Torczon (2000). Recently, Audet and Dennis Jr (2003) present a new and simple convergence analysis for the unconstrained and linearly constrained pattern search algorithms.

Pattern search methods have iterations made up of two parts: an *exploratory moves algorithm* and an algorithm for *updating of the pattern*. The exploratory moves algorithm has the aim of determining the best descent step $\mathbf{s}_k$ among a set of steps $\mathbf{s}_k^i$ obtained by evaluating the objective function at each trial point $\mathbf{x}_k^i = \mathbf{x}_k + \mathbf{s}_k^i$ according to the given search method, where $\mathbf{x}_k$ is the best trial point obtained at the (k-1)th iteration and i is each ith direction defined by the pattern. The updating of the pattern carries out contraction or expansion of the set of trial points in accordance with the pattern and the values of the objective function at each trial point $\mathbf{x}_k^i$. A generalized pattern search method for unconstrained minimization was introduced by Torczon (1997).

With respect to the number of function evaluations, the pattern search algorithms at least evaluate the objective function $2n$ times per iteration in the exploratory moves algorithm, which could be a disadvantage, specially, when the evaluation of the objective function is expensive. However, pattern search methods guarantee global convergence under some considerations of differentiability of the objective function (Torczon 1997) and (Audet and Dennis Jr 2003).

Through a simple example, a didactic explanation of pattern search methods is presented by Kolda et al. (2003).

2.2.3 The Nelder-Mead simplex method

The NM simplex method (Nelder and Mead 1965), which should not be associated with the widely known simplex algorithm of Dantzig for solving linear optimization problems, emerged from an original idea of Spendley et al. (1962), who proposed the use of a sequence of experimental simplex designs for obtaining an improvement of the objective function. The method of Spendley et al. (1962) computes the reflection of the worst vertex of the current simplex for finding a trial point that is compared with the vertices of the current simplex, so the method makes decisions for identifying a trial point better than the worst vertex at each iteration.

New operations, called expansion, contraction and shrinkage, were included by Nelder and Mead (1965) to the original optimization method of Spendley, Hext, and Himsworth, resulting in what nowadays is widely known as the NM simplex algorithm. We can remark that in most cases, the NM method carries out one function evaluation per iteration, because the NM method seldom executes shrink operations during the process of optimization. In fact, according to Lagarias et al. (1998), V. Torczon reported in her PhD thesis uniquely 33 shrinkage operations in 2.9 millions iterations on a set of test problems. Moreover, if the objective function is convex non-shrinkage operation takes place during the application of the NM method.

Since the NM method constitutes the essence of this research, we do not present more detail about it at this point. However, we should point out that the order of function evaluations of the NM method is one per iteration. This constitutes one of the most relevant and useful properties when developing our linearly constrained version of the algorithm. Another advantage that we find in the NM method is its good performance for unconstrained optimization problems when the objective function is altered by noise, for instance, (Barton and Ivey 1996) and (Humphrey and Wilson 2000).

Furthermore, an important number of numerical test problems reported by Rabinowitz (1995) for comparing the NM method with his called Torus algorithm, corroborate the potentiality of the NM method. According to the numerical results reported by Rabinowitz (1995), the NM method is considerably more efficient than the Torus algorithm, at least with regard to the number of function evaluations. However, the NM method seems to be less robust than the Torus algorithm, because the NM method eventually fails in more test problems than the method of Rabinowitz. We should point out that Rabinowitz (1995) compares his algorithm with the NM method using 64 of the test problems from Hock and Schittkowski (1981), and he also compares his method against both the NM method and the Simulated Annealing algorithms, whose numerical test problems are reported by Corana et al. (1987).

Nonetheless, in our opinion, the comparative study of Rabinowitz (1995) and Corana et al. (1987) should not be considered as conclusive, because they compare their optimization methods using constrained optimization problems versus the NM simplex method for finding solutions to unconstrained minimization problems, under the same objective function at each test problem.

On the other hand, Anderson and Ferris (2001) developed a method of direct search derived from the simplex and operations of the NM method for minimizing unconstrained problems when the objective function is subject to random noise. According to the computational results obtained by Anderson and Ferris (2001), their method is effective when the objective function contains noise.

2.2.4 Derivative-free methods

Historically, derivative-free methods had an important place in the decade of 60's. For instance, the method of Hooke and Jeeves (Bazaraa and Shetty 1979), Rosenbrock (1960), Powell (1964), among others. These are regarded as derivative-free methods, because they employ line search for minimizing an unconstrained non-linear objective function. We should remark that the method of Hooke and Jeeves presented in (Bazaraa and Shetty 1979) is not the original pattern search method of Hooke and Jeeves (1961). The method is so-called by Bazaraa and Shetty (1979), because it is founded on the original method of Hooke and Jeeves (1961).

Recently a derivative-free method was developed by Lucidi et al. (2002) for finding solutions to constrained minimization problems. The Lucidi-Sciandrone-Tseng (LST) method will be briefly described herein, due to its efficiency for minimizing linearly constrained non-linear objective function, according to the numerical test problems reported by Lucidi et al. (2002).

We were not able to consider fully the LST method in our exploration of approaches, because of its recent publication.

The LST method regards a constrained non-linear optimization problem of the form

$$\begin{array}{rl} & \min_{\mathbf{x}\in\mathbb{R}^n} f(\mathbf{x}) \\ \text{subject to:} & g_i(\mathbf{x}) \leq 0 \quad \forall i = 1,\dots,m \end{array}$$

where:

$f : \mathbb{R}^n \to \mathbb{R}$ and its first-order derivatives cannot be explicitly calculated.

$g_i : \mathbb{R}^n \to \mathbb{R}$ are continuously differentiable functions and their mathematical expressions are available.

The LST method defines at each kth iteration a set of feasible descent directions $D_k = \{\mathbf{d}_k^1, \dots, \mathbf{d}_k^{r_k}\}$, where the set of directions D_k is constituted by all normalized directions such that

$$\text{cone}\{D_k\} = T(\mathbf{x}_k, \epsilon_k) = \{\mathbf{d} \in \mathbb{R}^n \mid \nabla g_i(\mathbf{x})^T\mathbf{d} \leq 0 \ \forall i \in I(\mathbf{x}_k, \epsilon_k)\},$$

where $I(\mathbf{x}_k, \epsilon_k) = \{i = 1, \dots, m \mid g_i(\mathbf{x}_k) \geq -\epsilon_k\}$.

The LST method is essentially composed of three main steps at each kth iteration: selection of a set of directions D_k, minimization of the cone$\{D_k\}$ and search of the improvement of the function using the minimized cone.

Basically, the LST method identifies at each iteration, k, the best direction $\mathbf{d}_k^j \in D_k$ to use for a line search subject to the constraints.

Results of numerical examples reported by Lucidi et al. (2002) depict a good accuracy within an acceptable number of function evaluations.

A study of convergence of the LST method for unconstrained optimization is presented by Lucidi and Sciandrone (2002).

2.3 Response surface methodology

The optimization using response surface methodology (RSM) can be classified into two groups: metamodel and sequential procedures (Fu 1994). A metamodel procedure identifies the optimal solution by estimating the stationary point of the second-order response surfaces of the objective function (Myers and Montgomery 2002). These procedures employ experimental design for fitting a second-order response surfaces, and so to estimate the stationary point. Notice that according to the taxonomic criterion of descent direction of the objective function, these procedures would be within the group of method of zero order, because it does not employ the expansion of the objective function into Taylor series for estimating at least a descent direction.

The idea of optimizing a function through a sequential procedure for moving to a near stationary region could be attributed to Box and Wilson (1951), who proposed the response surface methodology (RSM) for fitting a first-order response surface, and so to estimate an ascent direct (or descent) for going towards a near stationary region at each iteration.

A sequential procedure is composed of two phases, the first is to estimate the first-order response surface of the objective function for finding a descent direction of the objective function, so moving towards a region near to the stationary point. The second phase computes the stationary point through the second-order response surfaces of the objective function.

These latter procedures are in the group of method of first-order, because they employ the first-order expansion Taylor series for identifying a descent direction at each iteration.

Recently, Fu (1994) proposed the use of a sequential procedure for identifying the optimum operation in simulation model, we present herein.

2.3.1 The sequential procedure method

Let $f(\mathbf{x}) : \mathbb{R}^n \rightarrow \mathbb{R}$ be a real function of several variables, the unconstrained optimization problem is given by

$$\min_{\mathbf{x} \in \mathbb{R}^n} f(\mathbf{x})$$

First phase of the sequential procedure

During the first phase, first-order experimental designs are applied at each kth iteration to obtain a hyperplane that fits to the response surface of the objective function, and whose analytical expression is given by

$$f(\mathbf{x}) \approx \hat{f}_k^{[1]}(\mathbf{x}) = \hat{\beta}_{0,k} + \mathbf{x}^T\hat{\boldsymbol{\beta}}_k = \hat{\beta}_0 + \sum_{i=1}^{n} \hat{\beta}_{i,k}x_i \quad \forall \mathbf{x} \in N_\varepsilon(\mathbf{x}_k),$$

where $N_\varepsilon(\mathbf{x}) = \{\mathbf{x} \mid \|\mathbf{x} - \mathbf{x}_k\| < \varepsilon\ \}$ is the ε-neighbourhood of $\mathbf{x}_k$ in a n-dimensional Euclidean space $\mathbb{R}^n$ for some $\varepsilon > 0$, $\hat{\beta}_{i,k}$ for all $i = 0, \ldots, n$ are the estimated coefficients via the least squares method at each kth iteration.

Therefore, a step descent direction can be estimated for approaching to a subregion R_s that contains at least a stationary point using

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \nabla \hat{f}_k^{[1]}(\mathbf{x}) = \mathbf{x}_k + \alpha_k \hat{\boldsymbol{\beta}}_k \quad \forall k = 0, \ldots,$$

where $\mathbf{x}_k$ is the point of the kth explored subregion, α_k is the step size whose value can be calculated by some line search method, and $\nabla \hat{f}^{[1]}(\mathbf{x})$ represents the gradient of the estimated hyperplane at kth iteration. This process of search is stopped by some criterion, which would allow one to reach a point $\mathbf{x}_K$ in R_s.

Second phase of the sequential procedure

In this phase, a set of second-order designs is carried out for fitting a quadratic response surface, and estimating the stationary point by first-order optimality condition.

$$f(\mathbf{x}) \approx \hat{f}^{[2]}(\mathbf{x}) = \hat{\beta}_0 + \mathbf{x}^T\hat{\boldsymbol{\beta}} + \mathbf{x}^T\hat{C}\mathbf{x} \quad \forall \mathbf{x} \in R_s,$$

where $\hat{\boldsymbol{\beta}}^T = [\beta_1, \ldots, \beta_n]$ and $\hat{C}$ is the $n \times n$ estimated symmetric matrix

$$\hat{C} = \begin{bmatrix} \hat{c}_{11} & \hat{c}_{12}/2 & \cdots & \hat{c}_{1n}/2 \\ & \hat{c}_{22} & \cdots & \hat{c}_{2n}/2 \\ & & \ddots & \vdots \\ \text{sym} & & & \hat{c}_{nn} \end{bmatrix}$$

By differentiating $\hat{f}^{[2]}(\mathbf{x})$ with respect to $\mathbf{x}$ and setting the derivative equal to 0, it is obtained the stationary point of the second-order model

$$\mathbf{x}_s = -\frac{1}{2}\hat{C}^{-1}\hat{\boldsymbol{\beta}},$$

which could be a minimum, maximum or saddle point (Myers and Montgomery 2002).

To verify if $\mathbf{x}_s$ is a minimum point we must study the matrix $\hat{C}$ by examining the signs of the eigenvalues of the matrix $\hat{C}$.

Let $\mu_1, \mu_2, \ldots \mu_n$ be the eigenvalues of $\hat{C}$, then

- If $\mu_1, \mu_2, \ldots \mu_n$ are all negative, $\mathbf{x}_s$ is considered a maximum point, due to the concavity of the function
- If $\mu_1, \mu_2, \ldots \mu_n$ are all positive, $\mathbf{x}_s$ corresponds to a minimum point, due to the convexity of the function
- If $\mu_1, \mu_2, \ldots \mu_n$ are mixed in sign, $\mathbf{x}_s$ is called a saddle point, because, the function is neither convex nor concave in the ε-neighbourhood of $\mathbf{x}_s$ (see definition in Appendix A).

As can be appreciated, the sequential procedure method requires n function evaluations at kth each iteration during the first phase, making it an expensive procedure due to its number of function evaluations per iteration. Nonetheless, the method could be efficient if the starting point is close to the stationary point.

2.4 The motivation of the Nelder-Mead method

In summary we seek a method that is economic from the standpoint of the number of function evaluations, within adequate accuracy, when the objective function is altered by the presence of noise.

The work presented by Brea (2002) indicates the advantages and disadvantages of the NP method, the NM simplex method and the RSM, in the minimization of a non-linear function subject to linear constraints, assuming that both the objective function and its first-order derivatives cannot be explicitly calculated or that they are computational expensive.

According to a comparative study made by Neddermeijer et al. (1999), the RSM was more accurate than the NM simplex method in the cases of optimization by simulation models. However for a set of 18 unconstrained test problems, whose objective functions were affected by noise the NM simplex method would be carried out more efficiently than the RSM for optimization cases by simulation models.

These considerations induced us to select an approach based on the NM simplex method, due to its acceptable performance, even when the objective function is corrupted by noise, for instance, (Humphrey and Wilson 2000). Another advantageous feature of the NM simplex method is minimum number of function evaluations per iteration, that in most situations is one function evaluation per iteration. Nonetheless, the NM method has shown to be a non-robust method, because its eventually could convergence to non-stationary point (McKinnon 1998).

In addition, the NM simplex method is easily adaptable to Euclidean sub-space that can be defined by the intersection of two or more linear constraint boundaries, when the simplex reaches the boundary of the feasible region. This feature allows us to adjust the simplex in the Euclidean sub-space by reducing of its number of vertices. The new algorithm therefore avoids the degeneration of the simplex during the identification of the optimum. Furthermore, the reduction of number of vertices can improve the efficiency of the new algorithm keeping the features of the NM method.

Another reason is the easy implementation that can have an algorithms of optimization based on the NM simplex method, because it is essentially defined by simple operations, which were adapted to our optimization problem. Nonetheless, the new algorithm logically required of additional procedures that were developed during the investigation.

Chapter 3

Linear Constrained Nelder-Mead method

3.1 Introduction

In the last few years, there has been a special interest in using the NM simplex method in searching for the optimal performance in stochastic systems. The idea of searching for the optimum operation of a system using the simplex was originally proposed by Spendley et al. (1962). They supported their method on the study of the simplex done by Box (1952), and Brooks and Mickey (1961), whose works showed that experimental designs based on the simplex are optima for estimating the slope of a noisy function. This property of the simplex design was taken advantage by Spendley et al. (1962) for developing an optimization method of noisy objective function. The method identifies the optimum point by estimating the reflection of the worst design point or vertex of each simplex, thereby obtaining a better vertex at each iteration.

An improvement to the method of Spendley et al. (1962) was formulated by Nelder and Mead (1965); (1966), including other operations to the simplex, called expansion, contraction and shrinkage. Nowadays this method is called the well-known NM simplex method, which was initially developed for optimizing unconstrained determinate non-linear objective functions. However, an extension of the NM method to constrained non-linear optimization problems was developed by Subrahmanyam (1989), who employed what he called a delayed reflection for preventing the collapse of the simplex onto the boundary of the feasible region. The Subrahmanyam method (SM) also makes use of a very simple penalty rule when a vertex becomes infeasible, due to an operation of reflection or expansion. Another modification of the NM method was developed by Hedlund and Gustavsson (1998), in which the authors apply the NM method to bounded constrained optimization using the method of Routh et al. (1977). This latter fits reflection and expansion trial points by correcting their coordinates, when the point is infeasible.

The most recent research published on the NM method for simulation optimization are, for instance, (Barton and Ivey 1996) and (Humphrey and Wilson 2000). The authors propose some modifications to the NM method for solving problems of optimization in simulation models.

In this chapter we shall describe a variant of the original NM method applied to the problems of optimization subject to linear constraints, which will be called the Linear Constrained Nelder-Mead (LCNM) method. The LCNM method takes advantage of the eventual collapse of the simplex onto the boundary of the feasible region for reducing the number of their vertices, so decreasing the number of function evaluations. This latter aspect can be important when the evaluation of the objective function is expensive.

The method builds a simplex in what we have called intersection space or sub-space. This allows us to build a simplex with the minimum number of vertices and therefore, minimizes the number of evaluations of the objective function.

This chapter is organized as follows. In Section 3.2 we describe the type of problem that will be studied. Definitions and notations are given in Section 3.3 for explaining the algorithm. Mathematical principles of the LCNM algorithm are presented in Section 3.4. In Section 3.5 are defined the basic operation of the LCNM algorithm. In Section 3.6 we describe our algorithm by flow chart and pseudo-codes. A set of experimentations is described for testing the LCNM method and comparing its performance with the method of Subrahmanyam (1989) in Section 3.7. In Section 3.8 are presented a few test problems reported by Lucidi et al. (2002) for comparing the LCNM method against the LST and the Solver of Microsoft® Excel spreadsheet. Furthermore, a study of computational effort of the algorithm is shown in Section 3.9. Finally, in Section 3.10 we present conclusions about our research.

3.2 Statement of the problem

Let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a non-linear continuous objective function, where $\mathbf{x} = [x_1, x_2, \ldots, x_d]^T$ is a column vector of components $x_1, \ldots, x_d$ in the d-dimensional Euclidean space, and consider the problem of minimizing $f(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{F}$, where

$\mathcal{F}$ denote a non-empty set of feasible points given by

$$\mathcal{F} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A\mathrm{x} \geq \mathrm{b}\}$$

A mathematical formulation of our problem can be expressed as follows:

Problem 3.1 ($\mathcal{P}$)

$$\min_{\mathrm{x} \in \mathbb{R}^d} f(\mathrm{x}) \tag{3.1}$$

subject to

$$\begin{array}{llll} a_{11}x_1 + & a_{12}x_2 + & \cdots + & a_{1d}x_d \geq & b_1 \\ a_{21}x_1 + & a_{22}x_2 + & \cdots + & a_{2d}x_d \geq & b_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{k1}x_1 + & a_{k2}x_2 + & \cdots + & a_{kd}x_d \geq & b_k \end{array} \tag{3.2}$$

This formulation is the deterministic case. We shall also consider its extension to problems of optimization for simulation of stochastic systems, where random noise is present in the objective function, due to the randomness of the modelled system. However we focus in this chapter on the above deterministic case.

3.3 Preliminaries

In this section, we present some definitions and notations that will be used in the explanation of the LCNM method.

3.3.1 Definitions

Definition 3.1 (Simplex) *We define a simplex in a d-dimensional Euclidean space as a set of different points* x_i *for all* $i = 1, \ldots, v$, *where* v *is the number of points or vertices of the simplex and each one is represented by a column vector of dimension* d. *A simplex can be represented in matrix notation as*

$$S_v^{[q]} = [\mathrm{x}_1 : \mathrm{x}_2 : \ldots : \mathrm{x}_v] = \begin{bmatrix} x_{11} & x_{12} & \ldots & x_{1v} \\ x_{21} & x_{22} & \ldots & x_{2v} \\ \vdots & \vdots & \ddots & \vdots \\ x_{d1} & x_{d2} & \ldots & x_{dv} \end{bmatrix}$$

where the superscript q represents the iteration counter as result of each simplex operation.

Note that in the above definition the number of vertices (v) is equal to $d+1$, when the simplex is in d-dimensional Euclidean space.

Definition 3.2 (Edge matrix of a simplex) *A pth edge matrix $E_p^{[q]}$ of a qth simplex $S_v^{[q]}$ in a d-dimensional Euclidean space is defined as a matrix of dimension $d \times (v-1)$ whose jth column represents the edge of $S_v^{[q]}$ between a referential vertex $\mathbf{x}_p$ and $\mathbf{x}_j$ for all $j = 1, 2, \ldots, v_q$ and $j \neq p$, thus*

$$E_p^{[q]} = [\mathbf{x}_1 - \mathbf{x}_p : \mathbf{x}_2 - \mathbf{x}_p : \ldots : \mathbf{x}_{j \neq p} - \mathbf{x}_p : \ldots : \mathbf{x}_v - \mathbf{x}_p] \ \forall p = 1, \ldots, v \tag{3.3}$$

Definition 3.3 (Entire simplex) *Let $R_c(E_p^{[q]})$ be the column rank of the pth edge matrix $E_p^{[q]}$. A simplex is said to be entire in a d-dimensional Euclidean space, if $min\Big[R_c(E_1^{[q]}), R_c(E_2^{[q]}), \ldots R_c(E_v^{[q]})\Big]$ is equal to d. That is, there exist at least $d = v - 1$ linearly independent vectors $\mathbf{x}_p - \mathbf{x}_{j \neq p}$ of the any pth edge matrix.*

Definition 3.4 (Opposite hyperface of a simplex to a vertex) *Let $S_v^{[q]}$ be a qth simplex of v vertices in the d-dimensional Euclidean space. The part of the simplex defined by all v vertices of the simplex except the vertex $\mathbf{x}_j$ is said to be a jth opposite hyperface of a simplex or the opposite hyperface of a simplex to a jth vertex . In other words, the jth opposite hyperface of the simplex $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ is given by*

$$H_j^{[q]} = \{\mathbf{x}_i \in S_v^{[q]} \mid i = 1, \ldots, v \ \wedge i \neq j\} \ \ \forall j = 1, \ldots, v, \tag{3.4}$$

hereinafter, the symbol $\wedge$ will represent "and".

Definition 3.5 (Remaining hyperface) *The part of the simplex that has been left after removing only one of its vertices is said to be the remaining hyperface.*

Based on the previous definition, if the vertex $\mathbf{x}_v$ were removed at a qth iteration, its remaining hyperface would be the opposite hyperface of the simplex $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ to the vertex $\mathbf{x}_v$, which is given by

$$H_v^{[q]} = \{\mathbf{x}_i \in S_v^{[q]} \mid i = 1, \ldots, v-1\}$$

Definition 3.6 (Collapsed simplex) *Consider the case of d-dimensional variables, thus,* $\mathbf{x} \in \mathbb{R}^d$*. A simplex of v vertices is said to be collapsed onto the boundary of an ith linear constraint, if the v vertices belong to the ith linear constraint boundary.*

Definition 3.7 (Active constraint) *An ith linear constraint is said to be active or activated by a simplex, if all its vertices belong to the ith linear constraint boundary. In other words, if the following equation is satisfied*

$$a_i^T[\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v] - b_i\mathbf{1}_v^T = \mathbf{0}_v^T \tag{3.5}$$

where $\mathbf{1}_v$ *and* $\mathbf{0}_v$ *are v-dimensional column vectors whose elements are all one and zero, respectively.*

Definition 3.8 (Collapse simplex degree) *Consider the case of d-dimensional variables. A simplex of v vertices is said to be collapsed in r degrees, if the v vertices belong to the boundaries of any r linearly independent linear constraints. Therefore these r linear constraints are activated by the collapsed simplex.*

Definition 3.9 (Minor simplex) *Consider the case of d-dimensional variables. A simplex* $S_v^{[q]}$ *is said to be a minor simplex or sufficiently defined on any r linearly independent linear constraint boundaries, if its number of vertices (v) is equal to d+1−r and* $min\left[R_c(E_1^{[q]}), R_c(E_2^{[q]}), \ldots R_c(E_v^{[q]})\right]$ *is equal to* $d - r$*.*

Definition 3.10 (Degenerate simplex) *A simplex of v vertices is said to be degenerate if all its vertices belong to at least one of its hyperfaces but the simplex has not activated any linear constraint boundary.*

Definition 3.7 should not be confused with the widely known definition of active constraint due to a point $\mathbf{x}_0$ or constraint activated by a point $\mathbf{x}_0$, where a constraint $g_i(\mathbf{x}) \geq 0$ is said to be activated by $\mathbf{x}_0$, if $g_i(\mathbf{x}_0) = 0$ (Gill et al. 1991). In this writing, when we indicate that a constraint is active, we refer to the constraint that has been activated by a simplex, otherwise we point out that the constraint has been activated by a point.

3.3.2 Notation

q = Iteration counter which is increased after each simplex operation and sorting of vertices.

s = Stage counter which is increased after each convergence stage.

d = The dimension of the space of our decision variables.

v = Number of vertices of the simplex.

ε_c = Tolerance for considering if the simplex has collapsed.

r = Degree of the collapsed simplex.

Δr = Variation of collapsed degree.

k = Number of linear constraints.

α = Coefficient of reflection.

β = Coefficient of contraction.

γ = Coefficient of expansion.

δ = Coefficient of shrinkage.

Δ = Coefficient of improvement.

ρ = Coefficient of gap.

τ = Parameter of step size of the simplex for building it.

n_{ac} = Number of active constraint, according to definition 3.7.

$finger(i)$ = Pointer that indicates the ranking of each vertex according to the value of the objective function at each vertex. $\mathbf{x}_{finger(1)}$ is the $\mathbf{x}$ for which the function is minimal, $\mathbf{x}_{finger(2)}$ is the $\mathbf{x}$ for which the function is the next to minimum, and so on.

$\mathbf{x}_i^{[q]} \equiv [x_{1,i}, \ldots, x_{d,i}]^T$ = It denotes the ith vertex in the current qth iteration, however, the superscript $[q]$ will be suppressed for simplifying the notation, that is, $\mathbf{x}_i^{[q]}$ will be denoted as $\mathbf{x}_i$ without losing its meaning.

$f_i \equiv f(\mathbf{x}_i)$ = It is the value of the objective function at the design point $\mathbf{x}_i$.

$\mathbf{x}_{\min}$ = The design point whose value function $f(\mathbf{x}_{\min}) = f_{\min}$ is the lowest in comparison to the rest of the value function at each vertex of the simplex.

$\mathbf{x}_{\max}$ = The design point whose value function $f(\mathbf{x}_{\max}) = f_{\max}$ is the highest in comparison to the rest of the value function at each vertex of the simplex.

$\mathbf{x}_{ntw}$ = The design point whose value function $f(\mathbf{x}_{ntw}) = f_{ntw}$ is the next to worst value function $f(\mathbf{x}_{\max})$.

$\tilde{}$ = The symbol tilde above a denoted matrix represents a not necessarily sorted matrix according to $f(\mathbf{x}_i)$. For example, $\tilde{S}_v$ represents a simplex of v vertices, but not necessarily in ascending order of $f(\mathbf{x}_i)$.

3.4 Basic properties of linearly constrained optimization

Here we shall enunciate a proposition for identifying the intersection space of a set of linear inequality constraints, whose choosing is based on the following proposition.

Proposition 3.1 *Let $\mathcal{X}$ be a non-empty open set in $\mathbb{R}^d$ and let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ and $l_i(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be an ith linear function given by $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ for all $i \in \mathcal{I}$, where $\mathcal{I}$ is the set of subscripts of the linear functions. Consider the Problem $\mathcal{P}$ of minimizing $f(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{X}$ and $l_i(\mathbf{x}) \geq 0$ for every $i \in \mathcal{I}$. Let $\mathbf{x}_{best}$ be a feasible point such that $\mathbf{x}_{best}$ lies on all active linear ith constraints that belong to a non-empty set $\mathcal{A}_p(\mathbf{x}_{best}) = \{i \in \mathcal{I} | l_i(\mathbf{x}_{best}) = 0\}$, and this point was obtained through convergence of some method of minimization, but that it is not necessarily an optimum point of Problem $\mathcal{P}$. If $f(\mathbf{x})$ is differentiable at $\mathbf{x}_{best}$ and $f(\mathbf{x})$ has an unknown local minimum $\mathbf{x}^*$ of Problem $\mathcal{P}$, such that $\mathcal{A}_p(\mathbf{x}_{best}) = \mathcal{A}_p(\mathbf{x}^*)$ and $\nabla f(\mathbf{x}_{best})^T \nabla f(\mathbf{x}^*) > 0$, then the unknown local minimum $\mathbf{x}^*$ of Problem $\mathcal{P}$ is located in the intersection space of the constraints activated by $\mathbf{x}_{best}$, such that*

$$\sum_{i \in \mathcal{A}_p(\mathbf{x}_{best})} \tilde{u}_i \nabla f(\mathbf{x}_{best})^T \mathbf{a}_i > 0, \tag{3.6}$$

where $\tilde{u}_i \geq 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*) = \mathcal{A}_p(\mathbf{x}_{best})$ are the Lagrange multipliers.

Proof. If $\mathbf{x}^*$ is a local minimum of Problem $\mathcal{P}$, then by Theorem A.6 (page 209), we have that $\tilde{u}_i \geq 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$

$$\nabla f(\mathbf{x}^*) - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \mathbf{a}_i = 0. \tag{3.7}$$

On the other hand, since $f(\mathbf{x})$ is differentiable at $\mathbf{x}_{best}$, from Equation (3.7), we obtain

$$\sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \nabla f(\mathbf{x}_{best})^T \mathbf{a}_i = \nabla f(\mathbf{x}_{best})^T \nabla f(\mathbf{x}^*),$$

which is greater than zero, because $\nabla f(\mathbf{x}_{best})^T \nabla f(\mathbf{x}^*) > 0$, therefore,

$$\sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \nabla f(\mathbf{x}_{best})^T \mathbf{a}_i > 0. \tag{3.8}$$

Due to Equation (3.8) and $\tilde{u}_i \geq 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*) = \mathcal{A}_p(\mathbf{x}_{best})$, we have

$$\sum_{i \in \mathcal{A}_p(\mathbf{x}_{best})} \tilde{u}_i \nabla f(\mathbf{x}_{best})^T \mathbf{a}_i > 0.$$

■

Notice that Proposition 3.1 could constitute a criterion for choosing the active linear constraints that contain the unknown local optimum $\mathbf{x}^*$ of Problem $\mathcal{P}$.

Solution 3.1 *Under Proposition 3.1, there exists a non-unique solution* $\nabla f(\mathbf{x}_{best})^T \mathbf{a}_i > 0$ *for all* $i \in \mathcal{A}_p(\mathbf{x}_{best})$, *such that, Equation (3.6) holds.*

Criterion 3.1 (Selection of active constraints) *Let* $\mathcal{A}_p(\mathbf{x}_{best})$ *be a set of linear inequality constraints activated by* $\mathbf{x}_{best}$. *Let* $\mathcal{A}_{\nabla f}$ *be a subset of* $\mathcal{A}_p(\mathbf{x}_{best})$ *such that the scalar product* $\nabla f(\mathbf{x}_{best})^T \mathbf{a}_i > 0$ *for all* $i \in \mathcal{A}_p(\mathbf{x}_{best})$. *Choose all linear inequality constraints of* $\mathcal{A}_{\nabla f}$ *for identifying an intersection space for which Condition 3.1 holds.*

Now we suppose that we have r active constraints of the k constraints given by Equation (3.2) as a result of applying Criterion 3.1 or another criterion for choosing the active linear constraints due to $\mathbf{x}_{best}$. This means within the context of the NM method, that all vertices of the current simplex converged to the point $\mathbf{x}_{best}$, which could be either a minimum point or not. Thus, we define $A_c = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ as the set of active constraint subscripts whose α_i element is a chosen linear inequality constraint from Equation (3.2).

Proposition 3.2 (Intersection space) *Let*

$$A_{r \times d} \mathbf{x}_{d \times 1} = \mathbf{b}_{r \times 1}, \tag{3.9}$$

be the system of equations defined by the linear active constraints $A_c = \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$ *which were chosen by Criterion 3.1 or another criterion, where* $rank(A_{r \times d}) = r$. *Then the intersection space that satisfies Equation (3.9) is defined by the set of points* $\mathbf{x} = [x_1, x_2, \ldots, x_d]^T$ *such that* r *components* x*'s of* $\mathbf{x}$ *are dependent on* $(d - r)$ *independent components* x*'s of* $\mathbf{x}$.

Proof. To prove Proposition 3.2, we transform Equation (3.9) by Gauss elimination method. We first order the equations according to the number of non-zero entries that each equation has. Then for each row, the rule of pivoting is based on the descendent order of the absolute values $|a_{\alpha_i,j}|$ in that row. Using the Gauss elimination method in Equation (3.9) and considering the matrix of pivoting, we obtain the equation system

$$\begin{bmatrix} \mathring{A}_D & \mathring{A}_I \end{bmatrix} \begin{bmatrix} \mathbf{x}_D \\ \mathbf{x}_I \end{bmatrix} = \mathring{\mathbf{b}}, \tag{3.10}$$

where the vector $\mathbf{x}_D$ represents the dependent components x's of $\mathbf{x}$, $\mathbf{x}_I$ identifies the considered independent components x's of $\mathbf{x}$, and $\mathring{A}_D$, $\mathring{A}_I$ and $\mathring{\mathbf{b}}$ represent in matrix notation the transformed coefficients during the process of Gauss elimination. This information can be obtained from the matrix of pivoting where the subscript jth of the elements $\mathring{a}_{ij}$ in $\mathring{A}_D$ identifies those variables that are considered dependent. Therefore each of them is allocated on the principal diagonal of the matrix $\mathring{A}_D$ of dimension $r \times r$. Thus,

$$\mathbf{x}_D = \mathring{A}_D^{-1} \left(\mathring{\mathbf{b}} - \mathring{A}_I \mathbf{x}_I \right), \tag{3.11}$$

which allows us to corroborate that $\mathbf{x}_D \in \mathbb{R}^r$ and $\mathbf{x}_I \in \mathbb{R}^{d-r}$. ■

Note that the matrix $\mathring{A}_D$ is non-singular, so using Equation (3.11) it be computed the points that belong to the intersection space by fixing the value of $\mathbf{x}_I$.

Furthermore, a particular study of optimality conditions for linearly constrained optimization of non-linear function is presented in Appendix A.

3.5 Simplex operations

The NM method has four basic operations called reflection, expansion, contraction and shrinkage. The sorted qth simplex $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ with the vertices: $\mathbf{x}_{\min} = \mathbf{x}_1$, $\mathbf{x}_{ntw} = \mathbf{x}_{v-1}$ and $\mathbf{x}_{\max} = \mathbf{x}_v$ is defined by evaluating the objective function at each jth vertex of the current qth simplex $\tilde{S}_v^{[q]}$ and sorting the vertices according to the value of the objective function at each vertex.

Reflection

The operation of reflection yields the reflection point $\mathbf{x}_{refl}$ which is defined as the projection of $\mathbf{x}_{\max}$ through the centroid $\mathbf{x}_{cen}$ of the remaining hyperface $H_v^{[q]}$, where $\mathbf{x}_{cen}$ obviously is the centroid point of the hyperface $H_v^{[q]} = \{\mathbf{x}_i \in S_v^{[q]} \mid i = 1, \ldots, v-1\}$.

This operation is computed by

$$\mathbf{x}_{refl} = (1+\alpha)\mathbf{x}_{cen} - \alpha\mathbf{x}_{\max} \tag{3.12}$$

where $\alpha = 1$ is the reflection coefficient of the NM method, and the centroid $\mathbf{x}_{cen}$ of the remaining hyperface $H_v^{[q]}$ is estimated by

$$\mathbf{x}_{cen} = \frac{1}{v-1}\sum_{i=1}^{v-1}\mathbf{x}_i \tag{3.13}$$

Expansion

The expansion point $\mathbf{x}_{\exp}$ is calculated by the projection of $\mathbf{x}_{refl}$ in the direction from $\mathbf{x}_{cen}$ towards $\mathbf{x}_{refl}$ and its expression is given by

$$\mathbf{x}_{\exp} = (1-\gamma)\mathbf{x}_{cen} + \gamma\mathbf{x}_{refl} \tag{3.14}$$

where $\gamma = 2$ is the expansion coefficient of the NM method.

Another expression for Equation (3.14) is obtained by plugging Equation (3.12) into Equation (3.14), so that

$$\mathbf{x}_{\exp} = (1+\alpha\gamma)\mathbf{x}_{cen} - \alpha\gamma\mathbf{x}_{\max} \tag{3.15}$$

Contraction

The operation of contraction, also called inside contraction, gives as result the denoted trial point $\mathbf{x}_{cont}$, whose coordinates are on the line segment $\mathbf{x}_{\max}$ and $\mathbf{x}_{cen}$. It can be calculated by

$$\mathbf{x}_{cont} = (1-\beta)\mathbf{x}_{cen} + \beta\mathbf{x}_{\max} \tag{3.16}$$

where $\beta = 0.5$ is the contraction coefficient of the NM method, that is conventionally used.

It is worthwhile mentioning that there also exists the so-called outside contraction operation. However, the outside contraction will not be employed by the LCNM method.

Shrinkage

Shrinkage is the reduction of the current simplex by moving each vertex of the current simplex towards the vertex $\mathbf{x}_{\min}$. This operation is computed by

$$\mathbf{x}_j = (1-\delta)\mathbf{x}_{\min} + \delta\mathbf{x}_j, \quad \forall j = 2, \ldots, v \tag{3.17}$$

where $\delta = 0.5$ is the shrinkage coefficient of the NM method that normally used.

3.6 Linear Constrained Nelder-Mead algorithm

An extension to the NM method was developed for finding the solution to optimization problems subject to linear inequality constraints, to ensure that all new simplex operations yield feasible trial points.

3.6.1 Procedures

To extend the NM method, we have added a set of new procedures to the original algorithm, which are presented herein.

Linear Constraint Procedure

Since each trial point is generated by two vertices of the simplex, it is possible to determine where it comes from and where it goes to. This information allows us to determine whether a generated new point $\mathbf{x}_{new}$ is feasible or not.

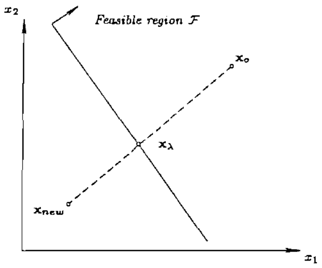


Figure 3.1: Linear Constraint Procedure.

In the case that the x_{new} does not satisfy Inequality (3.2), a procedure, called Linear Constraint Procedure (LCP), estimates the extreme trial point on the boundary of the feasible region as can be seen from Figure 3.1.

For this, the LCP computes the extreme feasible point $\mathbf{x}_\lambda$ for replacing the new point $\mathbf{x}_{new}$ by $\mathbf{x}_\lambda$, therefore, the infeasible new trial point $\mathbf{x}_{new}$ is converted to a new feasible trial point.

On the basis of this principle, we have the following propositions.

Proposition 3.3 (Constrained reflection operation) *Let $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ be the qth ranked simplex of v vertices such that all its vertices are in ascending order according to the value $f(\mathbf{x}_j)$ $\forall j = 1, \ldots .v$ and, all its vertices belong to the feasible region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$, given by k linear inequality constraints $\mathbf{a}_i^T\mathbf{x} \geq b_i$ $\forall i = 1, \ldots, k$. If a Nelder-Mead reflection operation takes place, which is computed by $\mathbf{x}_{refl} = (1+\alpha)\mathbf{x}_{cen} - \alpha\mathbf{x}_v$, then*

1) a constrained Nelder-Mead reflection operation of the current qth simplex is defined by

$$\mathbf{x}_{cnst\text{-}refl} = (1 + \min(\alpha, \lambda))\mathbf{x}_{cen} - \min(\alpha, \lambda)\mathbf{x}_v \tag{3.18}$$

where $\lambda = \min(\lambda_1, \ldots, \lambda_k)$ such that $\lambda_i \geq 0$ $\forall i = 1, \ldots, k$, and

$$\lambda_i = \frac{b_i - \mathbf{a}_i^T\mathbf{x}_{cen}}{\mathbf{a}_i^T(\mathbf{x}_{refl} - \mathbf{x}_{cen})}, \quad \forall i = 1, \ldots, k \tag{3.19}$$

2) If $0 \leq \lambda_i < 1$, then $\mathbf{x}_{refl}$ violates the inequality $\mathbf{a}_i^T\mathbf{x} \geq b_i$.

Proof. Part 1)

Let $\mathbf{x}_{refl}$ be a trial point which is computed by a Nelder-Mead reflection operation using

Equation (3.12). For the points $\mathbf{x}_{cen}$ and $\mathbf{x}_{refl}$, we consider the straight line L_{cr} that connects them using

$$\mathbf{x}_{\lambda_i} = \lambda_i \mathbf{x}_{refl} + (1 - \lambda_i)\mathbf{x}_{cen} \ \forall i = 1, \ldots, k \ , \tag{3.20}$$

where $\mathbf{x}_{\lambda_i}$ is the intersection point of the straight line L_{cr} and the boundary of the ith linear constraint, given by

$$\mathbf{a}_i^T \mathbf{x} = b_i \ \forall i = 1, \ldots, k \tag{3.21}$$

By substitution of $\mathbf{x}_{\lambda_i}$ given by Equation (3.20) into Equation (3.21), we obtain

$$\mathbf{a}_i^T [\lambda_i \mathbf{x}_{refl} + (1 - \lambda_i)\mathbf{x}_{cen}] = b_i \ \forall i = 1, \ldots, k \ .$$

Thus, we have

$$\lambda_i = \frac{b_i - \mathbf{a}_i^T \mathbf{x}_{cen}}{\mathbf{a}_i^T (\mathbf{x}_{refl} - \mathbf{x}_{cen})} \ \forall i = 1, \ldots, k \ ,$$

Note that the sign of λ_i allows us to determine whether the intersection point $\mathbf{x}_{\lambda_i}$ is located in the direction from $\mathbf{x}_{cen}$ towards $\mathbf{x}_{refl}$ or not. Since, the unique feasible intersection point $\mathbf{x}_{\lambda_i} \in \{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{a}_i^T \mathbf{x} = b_i \ \forall i = 1, \ldots, k\}$ located in the direction from $\mathbf{x}_{cen}$ towards $\mathbf{x}_{refl}$ that simultaneously satisfies all linear inequality constraints $\mathbf{a}_i^T \mathbf{x} \geq b_i$ $\forall i = 1, \ldots, k$, must be the closest to $\mathbf{x}_{cen}$ and whose $\lambda_j \geq 0$, then $\lambda = \min(\lambda_1, \ldots, \lambda_k)$, so

$$\mathbf{x}_{cnst\text{-}refl} = (1 + \min(\alpha, \lambda))\mathbf{x}_{cen} - \min(\alpha, \lambda)\mathbf{x}_v$$

Part 2) Because $\mathbf{x}_{\lambda_i}$ is a point that is located on the boundary of the ith linear constraint, we have

$$b_i = \mathbf{a}_i^T \mathbf{x}_{\lambda_i} \tag{3.22}$$

Plugging Equation (3.22) into Equation (3.19), we obtain for the ith linear constraint

$$\lambda_i = \frac{\mathbf{a}_i^T (\mathbf{x}_{\lambda_i} - \mathbf{x}_{cen})}{\mathbf{a}_i^T (\mathbf{x}_{refl} - \mathbf{x}_{cen})}$$

Obviously the angle between $\mathbf{a}_i^T$ and $(\mathbf{x}_{\lambda_i} - \mathbf{x}_{cen})$ is the same between $\mathbf{a}_i^T$ and $(\mathbf{x}_{refl} - \mathbf{x}_{cen})$, because $\mathbf{x}_{cen}$, $\mathbf{x}_{refl}$ and $\mathbf{x}_{\lambda_i}$ are collinear points, therefore

$$\lambda_i = \frac{||\mathbf{x}_{\lambda_i} - \mathbf{x}_{cen}||}{||\mathbf{x}_{refl} - \mathbf{x}_{cen}||} \tag{3.23}$$

Since $\mathbf{x}_{cen}$, $\mathbf{x}_{refl}$ and $\mathbf{x}_{\lambda_i}$ are collinear points, it can be clearly ascertained that if $\mathbf{x}_{\lambda_i}$ is located between $\mathbf{x}_{cen}$ and $\mathbf{x}_{refl}$, then $\mathbf{x}_{refl}$ violates inequality $\mathbf{a}_i^T\mathbf{x} \geq b_i$ and $0 \leq \lambda_i < 1$, because

$$\|\mathbf{x}_{\lambda_i} - \mathbf{x}_{cen}\| < \|\mathbf{x}_{refl} - \mathbf{x}_{cen}\| .$$

Otherwise, if $\mathbf{x}_{refl}$ is located between $\mathbf{x}_{cen}$ and $\mathbf{x}_{\lambda_i}$, then $\mathbf{x}_{refl}$ satisfies inequality $\mathbf{a}_i^T\mathbf{x} \geq b_i$ and $\lambda_i \geq 1$ as a result that

$$\|\mathbf{x}_{\lambda_i} - \mathbf{x}_{cen}\| \geq \|\mathbf{x}_{refl} - \mathbf{x}_{cen}\| .$$

■

Proposition 3.4 (Constrained expansion operation) *Let $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ be the qth ranked simplex of v vertices such that all its vertices are in ascending order according to the value $f(\mathbf{x}_j)$ $\forall j = 1, \ldots .v$ and, all its vertices belong to the feasible region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$, given by k linear inequality constraints $\mathbf{a}_i^T\mathbf{x} \geq b_i$ $\forall i = 1, \ldots, k$. If a Nelder-Mead expansion operation takes place, which is computed by $\mathbf{x}_{\exp} = (1 - \alpha\gamma)\mathbf{x}_{cen} - \alpha\gamma\mathbf{x}_v$, then*

1) a constrained Nelder-Mead expansion operation of the current qth simplex is defined by

$$\mathbf{x}_{cnst\text{-}\exp} = (1 + \min(\alpha\gamma, \lambda))\mathbf{x}_{cen} - \min(\alpha\gamma, \lambda)\mathbf{x}_v \tag{3.24}$$

where $\lambda = \min(\lambda_1, \ldots, \lambda_k)$ such that $\lambda_i \geq 0$ $\forall i = 1, \ldots, k$, and

$$\lambda_i = \frac{b_i - \mathbf{a}_i^T\mathbf{x}_{cen}}{\mathbf{a}_i^T(\mathbf{x}_{\exp} - \mathbf{x}_{cen})}, \quad \forall i = 1, \ldots, k$$

2) If $0 \leq \lambda_i < 1$, then $\mathbf{x}_{\exp}$ violates the inequality $\mathbf{a}_i^T\mathbf{x} \geq b_i$.

Proof. Parts 1) and 2) can be proved by the same method employed in the proof of Proposition 3.3. ■

Proposition 3.3 and Proposition 3.4 allow us to redefine the operations of reflection and expansion when the problem of minimizing a non-linear function is subject to inequality constraints. This is done in the next proposition.

Proposition 3.5 *Let $\mathbf{x}_o$ be a feasible point in the region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$ and let $\mathbf{x}_{new}$ be a new point generated by any simplex operation of the NM method, whose points* *define the line segment $L_{o,new}$. Let B_L be the subset of $\bar{k}$ ($0 \leq \bar{k} \leq k$) boundaries of $\mathcal{F}$ that are reached by the segment $L_{o,new}$. Also, let $\mathbf{x}_{\lambda_i}$ be the ith intersection point of the line segment $L_{o,new}$ with the ith boundary, expressed by*

$$\mathbf{a}_i^T \mathbf{x}_{\lambda_i} = b_i \quad \forall i \in B_L, \tag{3.25}$$

such that there uniquely exists a λ_i for each ith intersection point $\mathbf{x}_{\lambda_i}$, defined by

$$\mathbf{x}_{\lambda_i} = \lambda_i \mathbf{x}_{new} + (1 - \lambda_i)\mathbf{x}_o \qquad \forall 0 \leq \lambda_i \leq 1, \tag{3.26}$$

where λ_i is given by

$$\lambda_i = \frac{b_i - \mathbf{a}_i^T \mathbf{x}_o}{\mathbf{a}_i^T (\mathbf{x}_{new} - \mathbf{x}_o)}, \quad \forall i \in \mathcal{I} = \{\mathbb{N} \mid c_i \in B_L\} \tag{3.27}$$

Let Ξ denote the set of $\mathbf{x}_{\lambda_i}$ such that for each $\mathbf{x}_{\lambda_i}$ of Ξ Equation (3.25) and Equation (3.26) hold simultaneously. If the new point $\mathbf{x}_{new}$ is infeasible, that is $\mathbf{x}_{new} \notin \mathcal{F}$, then there only exists one feasible intersection point $\mathbf{x}_{\lambda_m} \in \Xi$, which is the closest to $\mathbf{x}_o$.

Proof. Since $\mathbf{x}_{new}$ is an infeasible point, by Equation (3.26) and Equation (3.25), we obtain all collinear points $\mathbf{x}_{\lambda_i}$, in the non-empty set Ξ, which can be sorted according to its distance to the point $\mathbf{x}_o$, and denoted by $\mathbf{x}_s$ for all $s \in \mathcal{I}$. Therefore, we obtain a ranked sequence of $\bar{k}$ points $\mathbf{x}_{\lambda_i} \in \Xi$, such that

$$\|\mathbf{x}_1 - \mathbf{x}_o\| \leq \|\mathbf{x}_2 - \mathbf{x}_o\| \leq \cdots \leq \|\mathbf{x}_{\bar{k}} - \mathbf{x}_o\|.$$

It is clear that the unique intersection point in Ξ, for which all inequality constraints hold is the point $\mathbf{x}_{s=1}$, which is also denoted by $\mathbf{x}_{\lambda_m}$, and is the point nearest to $\mathbf{x}_o$. ∎

Through Proposition 3.5 we establish the following general procedure for constraining new trial points to lie in the feasible region.

Algorithm 3.1 (Linear Constraint Procedure) *Start procedure*

Given $\mathbf{x}_o$, $\mathbf{x}_{new}$, *and Inequality (3.2)*

Do, *for all* $i = 1, \ldots, k$

if $\mathbf{x}_{new}$ *does not satisfy* $a_{i,1}x_1 + a_{i,2}x_2 + \cdots + a_{i,d}x_d - b_i \geq 0$

then *Compute λ_i using Equation (3.27) and find the new coordinate* *of the constrained* $\mathbf{x}_{new}$ *assigning to* $\mathbf{x}_{new} \leftarrow \lambda_i \mathbf{x}_{new} + (1 - \lambda_i)\mathbf{x}_o$

End do

End procedure

Observe that the reflection operation or expansion operation uniquely produces a new trial point outside of the current simplex, because the contraction operation employed by this variant of the NM method, also called inside contraction operation, generates a new trial point on the line segment defined by $\mathbf{x}_v$ and $\mathbf{x}_{cen}$ of the current simplex $S_v^{[q]}$.

With respect to the operation of shrinkage, it transforms the current simplex $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_v]$ by moving of each jth vertex $\mathbf{x}_j$ towards $\mathbf{x}_1$ along its current edge $\mathbf{x}_1 - \mathbf{x}_j$ for all $j = 2, \ldots, v$, which assures us that the set of new trial points belongs to the $(q+1)$th simplex, and they are on the edges of the previous $S_v^{[q]}$ simplex.

Feasible Entire Simplex Builder Procedure

This procedure defines the vertices of the biggest entire simplex inside of the feasible region given an initial interior vertex $\mathbf{x}_1$ of the feasible set. The following recurrent procedure builds an initial simplex S_0 which satisfies the constraints given by Equation (3.2). This procedure is founded on the generation of feasible points around the initial vertex and the parameter of maximum distance ν.

Algorithm 3.2 (Feasible Entire Simplex Builder Procedure) *Start procedure*

Given ν, $v = d + 1$ *and* $\mathbf{x}_1 = [x_{11}, x_{21}, \ldots, x_{d1}]^T$ *interior point*

Do, for all $j = 1, \ldots, v$, *assign to* $\mathbf{x}_j \leftarrow \mathbf{x}_1$

End do

Do, for all $j = 2, \ldots, v$

$x_{j-1,j} \leftarrow x_{j-1,j} - \nu$

Perform the ***Linear Constraint Procedure*** *to* $\mathbf{x}_j$

regarding that $\mathbf{x}_o = \mathbf{x}_1$ *and* $\mathbf{x}_{new} = \mathbf{x}_j$

$D^l = | x_{j-1,j} - x_{j-1,1} |$

$x_{j-1,j+1} \leftarrow x_{j-1,j+1} + \nu$

Perform the ***Linear Constraint Procedure*** *to* $\mathbf{x}_j$

regarding that $\mathbf{x}_o = \mathbf{x}_1$ *and* $\mathbf{x}_{new} = \mathbf{x}_{j+1}$

$D^r = | x_{j-1,j+1} - x_{j-1,1} |$

if $D^l > D^r$

then $x_{j-1,j+1} \leftarrow x_{j-1,1}$

else $x_{j-1,j} \leftarrow x_{j-1,j+1}$ *and* $x_{j-1,j+1} \leftarrow x_{j-1,1}$

End do

End procedure

Minor Simplex Builder Procedure

The procedure builds the biggest possible simplex in an intersection space given: an initial point $\mathbf{x}_1$ in the intersection space; the step size ν and the set of chosen active constraints that have been transformed by Proposition 3.2. The procedure computes the rest of the vertices required for defining a minor simplex in the intersection space. The procedure is a variant of the Feasible Entire Simplex Builder Procedure. A summary of the algorithm is as follows.

Algorithm 3.3 (Minor Simplex Builder Procedure) ***Start procedure***

Given ν *and an initial point* $\mathbf{x}_1 = [x_{11}, x_{21}, \ldots, x_{d1}]^T$ *in the intersection space.*

Given $I = \{\xi_1, \xi_2, \ldots, \xi_M\}$ *the set of index which represents the independent variable yielded by Equation (3.10).*

Do, *for all* $j = 1, 2, \ldots, v$, *where* $v = d + 1 - r$

$\mathbf{x}_j \leftarrow \mathbf{x}_1$

End do

$k \leftarrow 0$

Do, *for all* $j = 1, \ldots, v$

$k \leftarrow k + 1$

$x_{\xi_k,j} \leftarrow x_{\xi_k,j} - \nu$

Estimate the dependent components of $\mathbf{x}_j$ *using Equation (3.10).*

Perform the **Linear Constraint Procedure** *to* $\mathbf{x}_j$

regarding that $\mathbf{x}_o = \mathbf{x}_1$ *and* $\mathbf{x}_{new} = \mathbf{x}_j$

$D^l = \|\mathbf{x}_j - \mathbf{x}_1\|$

$x_{\xi_k,j+1} \leftarrow x_{\xi_k,j+1} + \nu$

Estimate the dependent components of $\mathbf{x}_{j+1}$ *using Equation (3.10).*

Perform the **Linear Constraint Procedure** *to* $\mathbf{x}_{j+1}$

regarding that $\mathbf{x}_o = \mathbf{x}_1$ *and* $\mathbf{x}_{new} = \mathbf{x}_{j+1}$

$D^r = \|\mathbf{x}_{j+1} - \mathbf{x}_1\|$

if $D^l > D^r$

then $\mathbf{x}_{j+1} \leftarrow \mathbf{x}_1$

else $\mathbf{x}_j \leftarrow \mathbf{x}_{j+1}$

$\mathbf{x}_{j+1} \leftarrow \mathbf{x}_1$

End do

End procedure

Because the LCNM algorithm takes advantage of when the simplex collapses onto the boundary of the feasible region through removing of vertices, it is necessary to establish a strategy for this removal substantiated by some properties of the simplex.

Definition 3.11 (Edge matrix of a remaining hyperface $H_{nv}^{[q]}$) *A pth edge matrix $E_{p,v}^{[q]}$ of a remaining hyperface $H_v^{[q]}$ in a d-dimensional Euclidean space, it is defined as a matrix of dimension $d \times (v-2)$ whose jth column represents the edge of the remaining hyperface $H_v^{[q]}$ of a sorted simplex $S_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_v]$, which is defined by subtraction between each vertex $\mathbf{x}_j$ and a referential vertex $\mathbf{x}_p$ for all $j = 1, 2, \ldots, v_q - 1$ and $j \neq p$, thus*

$$E_{p,v}^{[q]} = \left[\mathbf{x}_1 - \mathbf{x}_p : \mathbf{x}_2 - \mathbf{x}_p : \ldots : \mathbf{x}_{j \neq p} - \mathbf{x}_p : \ldots : \mathbf{x}_{v-1} - \mathbf{x}_p\right] \; \forall p = 1, \ldots, v-1 \qquad (3.28)$$

Theorem 3.4 *Let $R_c(E_{p,v}^{[q]})$ be the column ranks of pth edge matrices $E_{p,v}^{[q]}$ that can be defined by the vertices of the remaining hyperface $H_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_{v-2} : \mathbf{x}_{v-1}]$ at the* *qth iteration. Let*

$$R_c^{[q]} = \min\left[R_c(E_{1,v}^{[q]}), R_c(E_{2,v}^{[q]}), \ldots, R_c(E_{v-1,v}^{[q]})\right]$$

be the minimum of $R_c(E_{p,v}^{[q]})$*. Let* $\mathbf{x}_{cen}^{[q]}$ *be the current centroid point of the remaining hyperface* $H_v^{[q]}$ *of the current simplex* $S_v^{[q]}$*, where* $\mathbf{x}_{cen}^{[q]}$ *is different to any vertex of the current simplex* $S_v^{[q]}$*. If the vertex* $\mathbf{x}_{v-1}$ *of* $H_v^{[q]}$ *is replaced by* $\mathbf{x}_{cen}^{[q]}$*, then* $R_c^{[q+1]} \geq R_c^{[q]}$*, where*

$$R_c^{[q+1]} = \min\left[R_c(E_{1,v}^{[q+1]}), R_c(E_{2,v}^{[q+1]}), \ldots, R_c(E_{v-2,v}^{[q+1]}), R_c(E_{cen,v}^{[q+1]})\right]$$

of the (q+1)th edge matrices that can be defined by the not necessarily sorted hyperface $\tilde{H}_v^{[q+1]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_{v-2} : \mathbf{x}_{cen}]$.

Proof. By definition we have that the remaining hyperplane $H_v^{[q]}$ is given by

$$H_v^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_{v-2} : \mathbf{x}_{v-1}]$$

Since the vertex $\mathbf{x}_{v-1}$ of $H_v^{[q]}$ is replaced by $\mathbf{x}_{cen}^{[q]}$, the not necessarily sorted hyperface $\tilde{H}_v^{[q+1]}$ must have the same number of vertices as the hyperface $H_v^{[q]}$.

Let $\tilde{E}_{p,v}^{[q+1]}$ be the edge matrix constituted by the not necessarily sorted hyperface

$$\tilde{H}_v^{[q+1]} = [\mathbf{x}_1 : \mathbf{x}_2 : \ldots : \mathbf{x}_{v-2} : \mathbf{x}_{cen}^{[q]}], \tag{3.29}$$

where $\mathbf{x}_{cen}^{[q]} \notin \{\mathbf{x}_j \in H_v^{[q]} \mid j = 1, \ldots, v-1\}$ and is computed by

$$\mathbf{x}_{cen}^{[q]} = \frac{1}{v-1} \sum_{i=1}^{v-1} \mathbf{x}_i. \tag{3.30}$$

Case $p = 1, \ldots, v-2$ for $H_v^{[q]}$ and $\tilde{H}_v^{[q+1]}$.

If we consider the case where $p = 1, \ldots, v-2$, by hypothesis we obtain

$$\tilde{E}_{p,v}^{[q+1]} = \left[\mathbf{x}_1 - \mathbf{x}_p : \ldots : \mathbf{x}_{j\neq p} - \mathbf{x}_p : \ldots : \mathbf{x}_{v-2} - \mathbf{x}_p : \mathbf{x}_{cen}^{[q]} - \mathbf{x}_p\right]. \tag{3.31}$$

Letting $\mathbf{v}_j = \mathbf{x}_j - \mathbf{x}_p$ for all $p = 1, \ldots, v-2$, Equation (3.28) can be rewritten as

$$E_{p,v}^{[q]} = [\mathbf{v}_1 : \mathbf{v}_2 : \ldots : \mathbf{v}_{j\neq p} : \ldots : \mathbf{v}_{v-2} : \mathbf{v}_{v-1}] \;\; \forall p = 1, \ldots, v-2, \tag{3.32}$$

and for all $p = 1, \ldots, v-2$, from Equation (3.31), we obtain

$$\tilde{E}^{[q+1]}_{p,v} = \left[\mathbf{v}_1 : \mathbf{v}_2 : \ldots : \mathbf{v}_{j \neq p} : \ldots : \mathbf{v}_{v-2} : \mathbf{v}^{[q]}_{cen}\right]. \tag{3.33}$$

By Equation (3.30), we obtain

$$\mathbf{v}^{[q]}_{cen} = \frac{1}{v-1}\sum_{i=1}^{v-1}\mathbf{x}_i - \frac{1}{v-1}\sum_{i=1}^{v-1}\mathbf{x}_p = \frac{1}{v-1}\sum_{i=1}^{v-1}[\mathbf{x}_i - \mathbf{x}_p] \tag{3.34}$$

$$\mathbf{v}^{[q]}_{cen} = \frac{1}{v-1}\sum_{i=1}^{v-1}\mathbf{v}_i, \text{ where } \mathbf{v}_p = \mathbf{0}_d \tag{3.35}$$

Because $\mathbf{x}^{[q]}_{cen} \notin \{\mathbf{x}_j \in H^{[q]}_v \mid j = 1, \ldots, v-1\}$, then we can be assured that $\mathbf{v}^{[q]}_{cen} \neq \mathbf{v}_j$ for all $p \neq j = 1, \ldots, v-1$.

Since $\mathbf{v}^{[q]}_{cen}$ is different to all $\mathbf{v}_j$ for $j = 1, \ldots, v-1$ and some $p \neq j$, the column rank of $\tilde{E}^{[q+1]}_{p,v}$ must be equal to or more than the column rank of $E^{[q]}_{p,v}$.

Case $p = v-1$ for $H^{[q]}_v$ and $p = cen$ for $\tilde{H}^{[q+1]}_v$.

Now we study the particular case $E^{[q]}_{v-1,v}$ and $\tilde{E}^{[q+1]}_{cen,v}$, which can be written by

$$E^{[q]}_{v-1,v} = \left[\mathbf{x}_1 - \mathbf{x}_{v-1} : \mathbf{x}_2 - \mathbf{x}_{v-1} : \ldots : \mathbf{x}_j - \mathbf{x}_{v-1} : \ldots : \mathbf{x}_{v-2} - \mathbf{x}_{v-1}\right] \text{ and} \tag{3.36}$$

$$\tilde{E}^{[q+1]}_{cen,v} = \left[\mathbf{x}_1 - \mathbf{x}^{[q]}_{cen} : \mathbf{x}_2 - \mathbf{x}^{[q]}_{cen} : \ldots : \mathbf{x}_j - \mathbf{x}^{[q]}_{cen} : \ldots : \mathbf{x}_{v-2} - \mathbf{x}^{[q]}_{cen}\right]. \tag{3.37}$$

Since $\mathbf{x}_{v-1} = \mathbf{x}^{[q]}_{cen} + \mathbf{y}$, where $\mathbf{y} \neq \mathbf{0}_d$, Equation (3.37) can be rewritten as

$$\tilde{E}^{[q+1]}_{cen,v} = \left[\mathbf{x}_1 - \mathbf{x}_{v-1} + \mathbf{y} : \mathbf{x}_2 - \mathbf{x}_{v-1} + \mathbf{y} : \ldots : \mathbf{x}_j - \mathbf{x}_{v-1} + \mathbf{y} : \ldots : \mathbf{x}_{v-2} - \mathbf{x}_{v-1} + \mathbf{y}\right].$$

The above equation clearly allows us to affirm that the column rank of $\tilde{E}^{[q+1]}_{cen,v}$ is equal to the column rank of $E^{[q]}_{v-1,v}$.

By definition, we have

$$R_c^{[q]} \quad = \quad \min\left[R_c(E_{1,v}^{[q]}), R_c(E_{2,v}^{[q]}), \ldots, R_c(E_{v-2,v}^{[q]}), R_c(E_{v-1,v}^{[q]})\right]$$

$$R_c^{[q+1]} \quad = \quad \min\left[R_c(E_{1,v}^{[q+1]}), R_c(E_{2,v}^{[q+1]}), \ldots, R_c(E_{v-2,v}^{[q+1]}), R_c(E_{cen,v}^{[q+1]})\right]$$

Since $R_c(E_{p,v}^{[q+1]}) \geq R_c(E_{p,v}^{[q]})$ for all $p = 1, \ldots, v-2$ and $R_c(E_{cen,v}^{[q+1]}) = R_c(E_{v-1,v}^{[q]})$, then

$$R_c^{[q+1]} \geq R_c^{[q]}$$

■

Theorem 3.5 *Let $S_v^{[0]}$ be an initial feasible entire simplex of v vertices in the d-dimensional Euclidean space. Let $\mathcal{P}$ be the problem of minimizing an objective function $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ subject to $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$. Suppose that the NM algorithm is applied to the problem $\mathcal{P}$ with the constrained operations defined by Proposition 3.3 and Proposition 3.4. If after q iterations the sorted simplex $S_v^{[q+1]}$ defined by $[\mathbf{x}_1^{[q+1]} : \mathbf{x}_2^{[q+1]} : \cdots : \mathbf{x}_v^{[q+1]}]$ has at least all its vertices on a linear constraint boundary of the feasible region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$ except the vertex $\mathbf{x}_v^{[q+1]}$, which is an interior point of $\mathcal{F}$, and $f(\mathbf{x}_{cen}^{[q+1]}) \leq f(\mathbf{x}_v^{[q+1]})$ then the simplex will collapse onto the boundary of the feasible region at (q+1)th iteration by either a constrained reflection step or a constrained expansion step.*

Proof. Since at the beginning of the $(q+1)$th iteration the NM algorithm attempts a constrained reflection step, the worst vertex $\mathbf{x}_v^{[q+1]}$ is projected through $\mathbf{x}_{cen}^{[q+1]}$ by Equation (3.18). This clearly yields the constrained reflection trial point $\mathbf{x}_{cen}^{[q+1]}$, because

$$\mathbf{x}_{cnst\text{-}refl}^{[q+1]} = (1 + \min(\alpha, \lambda))\mathbf{x}_{cen}^{[q+1]} - \min(\alpha, \lambda)\mathbf{x}_v^{[q+1]} \text{ where } \lambda = 0.$$

Therefore $\mathbf{x}_{cnst\text{-}refl}^{[q+1]} = \mathbf{x}_{cen}^{[q+1]}$. Now the reflection step of the NM algorithm is:

If $f_{\min} \leq f_{refl} \leq f_{\max}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$ and go to the test of stopping

else go to expansion step.

Since $\mathbf{x}_{cnst\text{-}refl}^{[q+1]} = \mathbf{x}_{cen}^{[q+1]}$ then $f_{refl} = f_{cen}$. If $f_1 \leq f_{cen} \leq f_v$ then $\mathbf{x}_v^{[q+1]} \leftarrow \mathbf{x}_{cen}^{[q+1]}$ so the simplex collapses onto the boundary of the feasible region.

Otherwise, if $f_{cen} < f_1$ the NM algorithm attempts an expansion step.

In this case, it computes the expansion trial point by Equation (3.24), which yields again the centroid $\mathbf{x}_{cen}^{[q+1]}$, because

$$\mathbf{x}_{cnst\text{-}\exp}^{[q+1]} = (1+\min(\alpha\gamma,\lambda))\mathbf{x}_{cen}^{[q+1]} - \min(\alpha\gamma,\lambda)\mathbf{x}_{v}^{[q+1]} \text{ where } \lambda = 0.$$

Since $f_{cen} < f_1$, then $\mathbf{x}_v^{[q+1]} \leftarrow \mathbf{x}_{cen}^{[q+1]}$.

Because $\mathbf{x}_v^{[q+1]}$ is replaced by $\mathbf{x}_{cen}^{[q+1]}$ at the end of $(q+1)$th iteration, due to either reflection step or expansion step, the not necessarily sorted simplex $\tilde{S}_v^{[q+1]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_{v-1} : \mathbf{x}_{cen}]$ is a collapsed simplex. ■

From Theorem 3.4, Theorem 3.5 and Definition 3.9 (page 24), we have established the following criterion for reducing the number of vertices thus producing a collapse of the current simplex.

Criterion 3.2 (Removing vertices) *Let $S_v^{[q]}$ be a qth collapsed sorted simplex of v vertices in the d-dimensional Euclidean space, whose v vertices have activated r linear constraints. Then the minor simplex can be fitted by removing the r worst vertices from the $S_v^{[q]}$ collapsed sorted simplex.*

Applying Criterion 3.2, when a collapse of the simplex is produced onto r boundary linear constraints, can reduce the number of function evaluations, even when $\mathbf{x}_{cen}^{[q]}$ improbably belongs to $\{\mathbf{x}_j \in H_v^{[q]} \mid j = 1, \ldots, v-1\}$. Because of this, the removing of vertices often provides a minor simplex, in terms of Definition 3.9, using the best vertices according to objective function value.

3.6.2 Basic idea of the LCNM algorithm

The algorithm begins with a feasible point, which is used for building a feasible entire simplex. Given this initial simplex, the objective function $f(\mathbf{x}_i)$ is calculated at each ith vertex of the current simplex for sorting the vertices according to $f(\mathbf{x}_i)$.

Given $S_v^{[0]}$, a NM simplex operation is carried out, using the Linear Constraint Procedure (Algorithm 3.1) for changing any new infeasible trial point into a feasible trial point at each qth iteration.

This feasible trial point is evaluated using the objective function for deciding whether it replaces the vertex $\mathbf{x}_{\max}$ or not, using the logic of the NM method.

Because the simplex is transformed by some attempted LCNM simplex operations, it is necessary to check whether the transformed simplex has activated some linear constraint or not, in terms of Definition 3.7. In the case that the transformed current simplex activates r linear constraints, the algorithm fits the number of vertices (v) to $d+1-r$, by removing the r worst vertices from the collapsed sorted simplex. Thus, the LCNM algorithm would reduce the number of objective function evaluations.

The LCNM algorithm has two modes of operations: The first mode, called *saving mode*, does not evaluate the objective function at any expansion point $\mathbf{x}_{\exp}$, when the coordinates of $\mathbf{x}_{\exp}$ is equal to $\mathbf{x}_{refl}$, as a result of a constrained operation during a qth iteration. So the algorithm saves a function evaluation. The second mode is called *non-saving mode*, here the LCNM algorithm re-evaluates the function at the constrained expansion trial point, even when this has the same coordinates as the constrained reflection trial point during any qth iteration. Therefore the algorithm spends a function evaluation in this case. Though not efficient for the deterministic problem we could make good use of this re-evaluation when the objective function is affected by noise.

If the simplex converges onto the boundaries of r linear constraints at $\mathbf{x}_{opt}^{[q]}$, the algorithm identifies the active linear constraints whose scalar product $\nabla f(\mathbf{x}_{opt}^{[q]})^T \mathbf{a}_i$ are positive (see Condition 3.1 on page 26). Given these active constraints of positive scalar products, the algorithm builds a collapsed simplex in the intersection space of the identified constraints, in accordance with Equation (3.10). Thus when the algorithm is applied in the next $(s{+}1)$th stage again, it is now restricted to the intersection space.

When the number of active constraints with positive scalar product is greater or equal to the dimension d, the LCNM algorithm builds an entire simplex in the feasible region close to the convergent point $\mathbf{x}_{opt}^{[s]}$.

The algorithm makes use of two stopping rules. The first one is based on the maximal edge of the simplex, which is defined by $\Delta_{ij} = \max_{i \neq j} \|\mathbf{x}_i - \mathbf{x}_j\|$. If Δ_{ij} is less than some tolerance criterion $\eta > 0$, the algorithm considers as an optimum point the vertex $\mathbf{x}_1$ of the current simplex at the sth stage, which is denoted by $\mathbf{x}_{opt}^{[s]}$. The second stopping rule is based on $\left\|\mathbf{x}_{opt}^{[s]} - \mathbf{x}_{opt}^{[s-1]}\right\|$ and some $\Delta > 0$. If $\left\|\mathbf{x}_{opt}^{[s]} - \mathbf{x}_{opt}^{[s-1]}\right\| < \Delta$ the algorithm stops.

Furthermore, the algorithm moves to a next stage if the collapsed degree of the current simplex is more than one, otherwise the algorithm stops and assumes as optimum point the point $\mathbf{x}_{opt}^{[s]}$. Therefore further exploration will not be carried out in this case.

This latter criterion has the advantage of reducing additional exploration in the feasible region, because the algorithm does not restart again for doing more searches. Nonetheless, this criterion of stopping could guarantee the convergence to either the global minimum or a point that satisfies the sufficient conditions of Kuhn-Tucker (see

Appendix A).

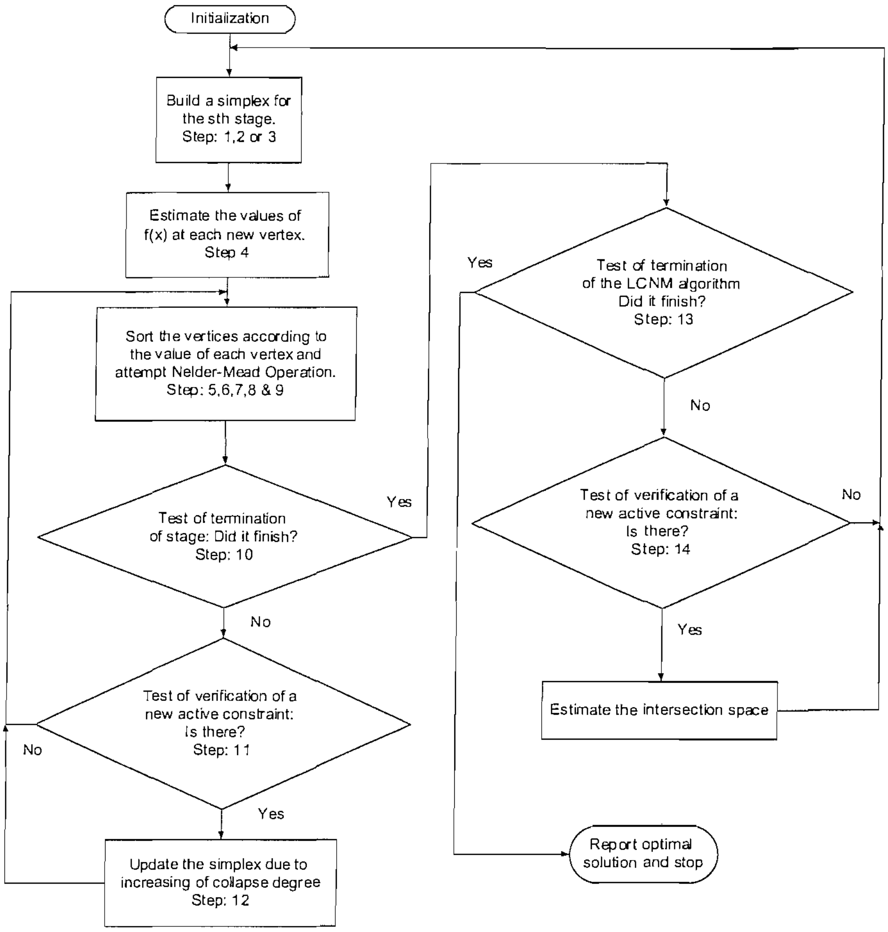


Figure 3.2: General flow chart of the LCNM algorithm.

Figure 3.2 depicts a basic flow chart of the LCNM algorithm, which points out each step of the algorithm that are explained by the pseudo-code algorithm of the following subsection. It is worthwhile mentioning that the LCNM algorithm employs Equation 3.38 for estimating the step size of each components, and Equation 3.39 as stopping rule. Both equations were taken from the developed NM method of Humphrey and Wilson (2000). Various initial simplices could be included in this algorithm, such as the initial simplex studied by Walters et al. (1991), who argue the advantages and disadvantages of each one.

Since Figure 3.2 is somewhat similar to Figure 1 from (Brea and Cheng 2003a), we have included the copyright holder's permission of Figure 1 in this work (see Appendix E on page 267).

3.6.3 The LCNM algorithm

For the purpose of describing the algorithm, we shall use pseudo-code notation, which will help in its implementation in any programming language.

Start

Initialization

Given a feasible initial point $\mathbf{x}_{initial}$ and therefore its dimension d.

Given the parameters of the Nelder-Mead method

$\alpha = 0.95$, $\beta = 0.5$, $\gamma = 2$ and $\delta = 0.5$.

Given the parameters of stopping rule $\eta_1 = \eta_2$.

Given the parameter of gap $\rho = 0.99$ and the coefficient of improvement $\Delta = 10\eta_1$.

Initialize

The number of vertices $v = d + 1$ and the stage $s \leftarrow 0$.

$\mathbf{x}_1 \leftarrow \mathbf{x}_{initial}$, $f(\mathbf{x}_{opt}^{[s]}) \leftarrow f(\mathbf{x}_1)$ and

$flag_{\mathbf{x}_1} \leftarrow true$, which indicates if $f(\mathbf{x}_1)$ is known.

$flag_{intet} \leftarrow false$, which indicates whether the current simplex was built from the intersection space or not.

The set of active constraints $S_{active} \leftarrow \varnothing$

The number of active constraints $n_{ac} \leftarrow 0$

Estimate

$$\nu = \begin{cases} \max\{\tau\,|x_{i,1}| : i = 1, 2, \ldots d\} & if\ \mathbf{x}_1 \neq \mathbf{0}_d, \\ 1 & otherwise \end{cases} \tag{3.38}$$

where $0.1 \leq \tau \leq 6$ is the step size parameter.

Go to step 1.

Step 1. Build an initial entire feasible simplex.

Perform the **Feasible Entire Simplex Builder Procedure** based on $\mathbf{x}_1$ for estimating the rest of the v feasible vertices of the simplex.

Go to step 4.

Step 2. Build a entire simplex next to $\mathbf{x}_{opt}^{[s]}$

Assign to $\mathbf{x}_1 \leftarrow \rho\mathbf{x}_{opt}^{[s]} + (1-\rho)\mathbf{x}_{initial}$ and to $v \leftarrow d+1$

Perform the **Feasible Entire Simplex Builder Procedure** based on $\mathbf{x}_1$ for estimating the rest of the v feasible vertices of the simplex.

Assign to $\mathbf{x}_1 \leftarrow \mathbf{x}_{opt}^{[s]}$ and to $flag_{\mathbf{x}_1} \leftarrow true$

Go to step 4.

Step 3. Build a simplex in the intersection space.

Given the vertex $\mathbf{x}_1$ and the component x's of $\mathbf{x}_I$ and $\mathbf{x}_D$, whose relationship is given by Equation (3.11).

Perform the **Minor simplex Builder Procedure.**

Assign to $flag_{intet} \leftarrow true$ and to $flag_{\mathbf{x}_1} \leftarrow true$

Go to step 4.

Step 4. Estimate the values of the vertices.

Given $flag_{\mathbf{x}_1} \in \{true, false\}$

Select the case according to $flag_{\mathbf{x}_1}$

Case *true*:

In this case, we know the value of $f(\mathbf{x}_1)$

Estimate $f_j = f(\mathbf{x}_j) \quad \forall \quad j = 2, 3, \ldots, v$

Case *false*:

This case is performed when we know the value of $f(\mathbf{x}_{\min})$ due to a shrinkage.

Estimate $f_j = f(\mathbf{x}_j) \quad \forall \quad j = 1, 2, \ldots, v$ and $j \neq \min$

End select

Assign to $flag_{\mathbf{x}_1} \leftarrow false$ and go to step 5.

Step 5. Attempt Constrained Reflection.

Sort $f_j \quad \forall \quad j = 1, \ldots, v$ for determining $\mathbf{x}_{\min}$, $\mathbf{x}_{ntw}$ and $\mathbf{x}_{\max}$.

Compute $\mathbf{x}_{cen}$ and $\mathbf{x}_{refl}$.

Perform the **Linear Constraint Procedure** to $\mathbf{x}_{refl}$ considering that

$\mathbf{x}_o = \mathbf{x}_{\max}$ and $\mathbf{x}_{new} = \mathbf{x}_{refl}$.

Estimate $f_{refl} = f(\mathbf{x}_{refl})$

if $f_{\min} \leq f_{refl} \leq f_{ntw}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$ and go to step 10 to test the stopping rule.

else go to step 6.

Step 6. Attempt Constrained Expansion.

if $f_{refl} \leq f_{\min}$

then Compute $\mathbf{x}_{\exp}$

Perform the **Linear Constraint Procedure** to $\mathbf{x}_{\exp}$ considering that

$\mathbf{x}_o = \mathbf{x}_{refl}$ and $\mathbf{x}_{new} = \mathbf{x}_{\exp}$.

if $f_{\exp} \leq f_{\min}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{\exp}$, $f_{\max} \leftarrow f_{\exp}$

else $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$

Go to step 10

else go to step 7

Step 7. Attempt Contraction

if $f_{refl} > f_{ntw}$

then Reduce the size of the current simplex, either by a contraction or by shrinking, by verifying the following.

if $f_{refl} \leq f_{\max}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$

Sort $f_j \quad \forall \quad j = 1, 2, \ldots, v$ for determining $\mathbf{x}_{\min}$, $\mathbf{x}_{ntw}$ and $\mathbf{x}_{\max}$.

Compute the contraction point $\mathbf{x}_{cont}$ and its value f_{cont} and go to step 8.

Step 8. Contract the simplex in one direction.

if $f_{cont} \leq f_{\max}$

then $\mathrm{x}_{\max} \leftarrow \mathrm{x}_{cont}$, $f_{\max} \leftarrow f_{cont}$ and go to step 10.

else go to step 9.

Step 9. Shrink current simplex.

if $f_{cont} > f_{\max}$

then Compute the new vertices through

$$\mathrm{x}_j = (1-\delta)\mathrm{x}_{\min} + \delta \mathrm{x}_j, \quad \forall j = 1, 2, \ldots, v$$

Assign to $flag_{\mathrm{x}_1} \leftarrow false$ and go to step 10.

else go to step 10.

Step 10. Test of termination criterion I

Sort $f_j \quad \forall \quad j = 1, 2, \ldots, v$ for determining $\mathrm{x}_{\min}$.

Compute the following inequality for the current simplex

$$\max_{\mathrm{x}_j} \|\mathrm{x}_j - \mathrm{x}_{\min}\| \leq \begin{cases} \eta_1 \|\mathrm{x}_{\min}\| & if \ \ \|\mathrm{x}_{\min}\| > \varepsilon, \\ \eta_2 & otherwise \end{cases} \tag{3.39}$$

where ε is a very small non-negative number.

if Inequality (3.39) is satisfied

then go to step 13, for keeping the optimum solution at $(s+1)$th stage and testing stopping rule

else go to step 11.

Step 11. Test for estimating a new active constraint.

$\Delta r \leftarrow 0$

Do, for all $i = 1, 2, \ldots, k \wedge i \notin S_{active}$

if $a_i^T[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_v] - b_i \mathbf{1}_v^T = \mathbf{0}_v^T$

then $\Delta r \leftarrow \Delta r + 1$ and identify the current linear constraint $l_i(\mathbf{x})$,

as active constraint. Thus, $S_{active} \leftarrow S_{active} \cup \{i\}$

End do

if $\Delta r > 0$ **then** go to step 12, **else** go to step 5.

Step 12. Update the simplex due to increasing of collapsed degree.

$v \leftarrow v - \Delta r$

Apply Criterion 3.2 by choosing the v best vertices from the current simplex.

Go to step 5.

Step 13. Test of termination criterion II

Change the stage s, therefore assign to $s \leftarrow s + 1$

$\mathbf{x}_{opt}^{[s]} \leftarrow \mathbf{x}_{\min}$, $f(\mathbf{x}_{opt}^{[s]}) \leftarrow f(\mathbf{x}_{\min})$

if $\left\| \mathbf{x}_{opt}^{[s]} - \mathbf{x}_{opt}^{[s-1]} \right\| < \Delta$

then go to step 16.

else go to step 14.

Step 14. Test for estimating a new active constraint.

Assign to $\mathbf{x}_{opt}^{[s]} \leftarrow \mathbf{x}_{opt}^{[s-1]}$ and $\Delta r \leftarrow 0$

Do, for all $i = 1, 2, \ldots, k \wedge i \notin S_{active}$

if $a_i^T[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_v] - b_i \mathbf{1}_v^T = \mathbf{0}_v^T$

then $\Delta r \leftarrow \Delta r + 1$ and identify the current linear constraint $l_i(\mathbf{x})$,

as active constraint. Thus, $S_{active} \leftarrow S_{active} \cup \{i\}$

End do

if $\Delta r > 0$ **then** go to step 15, **else** go to step 16.

Step15. Choosing of linear constraint for estimating the intersection space

Compute $\nabla f(\mathbf{x}_{opt}^{[s]})$ and assign to $S_{inter} \leftarrow \varnothing$ and

the number of active constraints $n_{ac} \leftarrow 0$

Do, for all $i = 1, 2, \ldots, k$

if $i \in S_{active}$, in other words, if $l_i(\mathbf{x})$ is activated by $\mathbf{x}_{opt}^{[s]}$

then Estimate the scalar product of $\left\langle \nabla f(\mathbf{x}_{opt}^{[s]})^T, \mathbf{a}_i \right\rangle$

if $\left\langle \nabla f(\mathbf{x}_{opt}^{[s]})^T, \mathbf{a}_i \right\rangle > 0$ (Criterion 3.1)

then Consider the current constraint for

estimating the intersection space using Proposition 3.2.

Thus, $S_{inter} \leftarrow S_{inter} \cup \{i\}$ and $n_{ac} \leftarrow n_{ac} + 1$

End do

if $S_{inter} \neq \varnothing$ and $n_{ac} < d$

then Estimate the intersection space and transform it according to Equation (3.10)

for identifying $\mathbf{x}_I$ and $\mathbf{x}_D$ and go to step 3.

else go to step 2.

Step 16. Report optimal solution

Report $\mathbf{x}_{opt}^{[s]}$ and $f(\mathbf{x}_{opt}^{[s]})$ as a local minimum solution and **stop** the algorithm.

3.7 Numerical experiments

In this section, we consider two groups of tests. The first has two experiments for showing preliminary computational results of the LCNM method. The second group is composed of five experiments for comparing the performance of the LCNM method, its variants that will be explain herein, and the method of Subrahmanyam (1989). We must indicate that the step size parameter τ was fixed to 0.2 for all experiments and the LCNM methods operated in saving mode.

It is worthwhile mentioning that the LCNM method displayed a good performance in these groups of experiments. However, in Chapter 5 we shall report some numerical test problems where the LCNM method fails.

3.7.1 Preliminary tests of the LCNM method

Experiment 1

Quadratic objective function

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2 \quad \text{where } 2 \leq d \leq 8$$

subject to:

$$\begin{aligned} 3x_1 + 2x_2 &\geq 120 \\ x_1 + 2x_2 &\leq 20 \end{aligned}$$

Considering as initial point $\mathbf{x}_{initial} = (400, -400, \underbrace{400, \ldots, 400}_{(d-2)\text{ times}})^T$ and the stopping parameters $\eta_1 = \eta_2 = 10^{-6}$. Observe that the local optimum solution is given by $\mathbf{x}_{\min} = (50, -15, \underbrace{0, 0, \ldots, 0}_{(d-2)\text{ times}})^T$ with $f(\mathbf{x}_{\min}) = 2725$.

Though the objective function has only one global minimum, this numerical example has a special interest in the study of performance of the algorithm when only x_1 and x_2 are restricted. The difficulty of convergence for this class of problem will be presented later on, where the Subrahmanyam method (SM) does not work correctly (see Experiment 4a on page 55).

Table 3.1: Summary of Experiment 1.

$f(\mathbf{x}_{\min})$	2725	2725	2725	2725	2725	2725.0011	2725.0061
d	2	3	4	5	6	7	8
NE	22	223	361	890	850	2316	6090
DTP	7.69E-14	5.93E-6	1.53E-5	1.58E-5	1.97E-5	2.86E-5	8.38E-5
x_1	50	50	50	50	50	50.000008	50.000042
x_2	-15	-15	-15	-15	-15	-15.000011	-15.000063
x_3		-5.93E-6	1.52E-5	-6.78E-6	-1.59E-5	-2.08E-5	1.53E-5
x_4			-1.67E-6	1.42E-5	-1.04E-6	7.72E-6	1.90E-5
x_5				1.13E-6	2.60E-6	-6.43E-6	1.04E-5
x_6					-1.14E-5	9.53E-6	2.40E-6
x_7						-9.77E-7	1.49E-5
x_8							1.81E-5

Table 3.1 depicts a summary of the designed experimentation for different dimensions. From this table, we can see that the LCNM method adequately converged to the optimum in all cases. We can also observe that the number of function evaluations (NE) and the distance to the true point (DTP) increased with the dimension of the problem.

Experiment 2

Extended Rosenbrock objective function

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2\right]$$

subject to:

$$\begin{aligned} x_1 + x_2 &\geq 10 \\ -2x_1 + x_2 &\leq 20 \\ -4x_1 + x_2 &\leq 0 \\ x_3 + x_4 &\geq 12 \quad \text{(for the case } d = 4 \text{ and } 6\text{)}, \end{aligned}$$

where d is an even integer number.

The parameters of stopping rule were fixed at $\eta_1 = \eta_2 = 10^{-6}$, and as initial point for each d-dimensional test was considered $\mathbf{x}_{initial} = 20 \cdot \mathbf{1}_d$, where $\mathbf{1}_d$ is the d-dimensional column vector, whose components are equal to 1.

Some features of the Extended Rosenbrock function are presented in Section 5.4.1, where we indicate more fully why this function was chosen for our numerical problem. Furthermore, the constraints that are only described in terms of x_1 and x_2 present two corners at $(2,8)^T$ and $(5,20)^T$ which are collinear to the true point $(4,16)^T$. This situation can cause the method to converge one of the corners.

Table 3.2: Summary of Experiment 2.

Result	LCNM	TP	LCNM	TP	LCNM	TP
$f(\mathbf{x}_{\min})$	8.994373	9	12.993557	13	12.993536	13
d	2	2	4	4	6	6
α	0.95		0.95		0.90	
NE	165		494		3337	
DTP	0.007734		0.007753		0.0080915	
x_1	3.998124	4	3.998124	4	3.998134	4
x_2	15.992497	16	15.992499	16	15.992543	16
x_3			2.999592	3	2.999597	3
x_4			9.000409	9	9.000403	9
x_5					1.001088	1
x_6					1.002211	1

Table 3.2 displays the results obtained by our method at two values of α and the theoretical optimum solution for $d = 2, 4$ and 6 indicated by the true point (TP). As can be seen from the table, the cases where all components x_i were constrained by the inequality constraints, the method approached TP with a fairly small number of function evaluations (NE). However, in the case $d = 6$ the components x_5 and x_6 were not constrained by any inequality, so the LCNM method required a large NE.

3.7.2 Comparative tests among the LCNM methods and Subrahmanyam method

The performance of the LCNM method and some of its variations versus the SM are compared herein. The variations of the LCNM method are based on bypassing the reduction of vertices when a new linear constraint has been activated by the current simplex or the building of a simplex in the intersection space. These variations of the LCNM method will be denote as follows,

Table 3.3: Notation of the variant of the LCNM method.

LCNM	Original version of the method.
LCNM-G	LCNM method without using the building of a simplex in the intersection space.
LCNM-R	LCNM method without reducing simplex vertices when it has activated a new constraint.
LCNM-RG	LCNM method without building of simplex in the intersection space and reducing of vertices.

With respect to the LCNM-G and the LCNM-RG, the methods build an entire simplex close to the optimum point $\mathbf{x}_{opt}^{[s]}$.

We apply the same procedure for creating an entire feasible simplex inside of the feasible region for all the methods, including the SM. Hence, all the methods begin with the same initial entire simplex.

Experiment 3

Comparison using quadratic objective function subject to a constraint

Here we shall show a comparison of the LCNM method, its variations and the SM for a convex quadratic objective function problem subject to a linear constraint. This experiment aims to illustrate the performance of the LCNM methods and the SM, when the objective function is strictly convex and a symmetric linear constraint.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2,$$

subject to $\sum_{i=1}^{d} x_i \geq l$ where $l = 12$.

The optimum point is $\mathbf{x}_{\min} = \frac{l}{d} \cdot \mathbf{1}_d$. For the case $d = 7$, the TP could not be set exactly but was set at $1.71143 \cdot \mathbf{1}_7$ which includes a rounding error. This affected the reported DTP values in this case only.

The parameters of stopping rule were fixed at $\eta_1 = \eta_2 = 10^{-6}$ for the LCNM methods, whilst the stopping rule setting of the SM were $\eta_1 = \eta_2 = 10^{-4}$ for 2 and 3 dimensions, otherwise $\eta_1 = \eta_2 = 10^{-3}$. On the other hand, the initial point for each d-dimensional test was considered $\mathbf{x}_{initial} = 20 \cdot \mathbf{1}_d$.

Table 3.4: Summary of Experiment 3.

d		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
	α	0.95	0.95	0.95	0.95	0.8
2	NE	97	**56**	114	73	268
	DTP	2.17E-6	2.58E-6	8.34E-7	8.33E-7	6.96E-4
	α	0.95	0.95	0.95	0.95	0.8
3	NE	196	**115**	229	148	442
	DTP	1.72E-6	3.09E-6	2.3E-6	2.3E-6	2.47E-2
	α	0.95	0.95	0.95	0.95	0.8
4	NE	382	**217**	561	258	613
	DTP	2.11E-6	2.11E-6	1.33E-6	1.33E-6	2.33E-2
	α	0.95	0.95	0.95	0.95	0.8
5	NE	571	**336**	628	383	663
	DTP	2.26E-6	3.51E-6	2.11E-6	2.11E-6	1.47E-2
	α	0.95	0.95	0.95	0.95	0.8
6	NE	1180	**511**	1244	571	765
	DTP	1.82E-6	2.02E-6	1.78E-6	1.78E-6	8.94E-1
	α	0.95	0.95	0.95	0.95	0.85
7	NE	1977	1977	2151	2151	**1418**
	DTP	1.17E-5	1.17E-5	1.17E-5	1.17E-5	1.79E-1
	α	0.95	0.95	0.95	0.95	0.85
8	NE	2104	**1144**	2282	1316	3043
	DTP	1.89E-6	1.89E-6	1.92E-6	3.26E-6	4.86E-1

Table 3.4 shows the results using the family of LCNM methods and the SM. From this table, we observe that the best performance was given by the LCNM-G method, because it satisfactorily found the optimum within a minor number of function evaluations (NE) for most cases.

Experiment 4

Comparison using quadratic objective function subject to several constraints

In this group of experiments, the family of the LCNM methods and the SM are compared using the following objective function

$$f(\mathbf{x}) = \sum_{i=1}^{d} x_i^2 \quad \text{where } \mathbf{x} \in \mathbb{R}^d$$

A set of C_i linear constraints defined herein, was employed in this group of experiments

$$\begin{array}{lrcl} C_1: & 3x_1 + 2x_2 & \geq & 120 \\ C_2: & x_1 + 2x_2 & \leq & 20 \\ C_3: & x_1 + x_2 + x_3 + x_4 & \geq & 80 \\ C_4: & x_1 + x_2 + x_3 + x_4 + 2x_5 + 2x_6 & \geq & 100 \\ C_5: & 2x_1 + 2x_2 + x_3 + x_4 + 2x_5 + 2x_6 & \geq & 135 \end{array} \qquad (3.43)$$

The parameter of reflection α was fitted for finding the best performance of each method, and the stopping parameters were fixed at $\eta_1 = \eta_2 = 10^{-6}$.

We have chosen these linear constraints, because the purpose of this experiment is to measure the performance of the methods, when non-redundant constraints are incorporated to the minimization of the same objective function. In this way, we shall comparatively study the methods when they operate under different circumstances. In addition, note that the constraints C_1 and C_2 form a sharp angle, which can cause a large number of evaluations. The set of constraints produces a feasible region with several sub-spaces that as be verified by the combination of the constraints. To perform satisfactorily the algorithms may therefore have to search some of these sub-spaces

Experiment 4a

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$$

subject to C_1 and C_2 from (3.43).

The optimum point is $\mathbf{x}_{\min}^T = (50,\ -15,\ \underbrace{0, \ldots, 0}_{(d-2) \text{ times}}\ )$ and as initial point was considered the point $\mathbf{x}_{initial} = (400,\ -400, \underbrace{400, \ldots, 400}_{(d-2) \text{ times}})$.

Table 3.5: Summary of Experiment 4a.

d		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
2	α	0.95	0.95	0.95	0.95	0.95
	NE	22	**17**	24	19	270
	DTP	7.69E-14	7.69E-14	2.05E-13	2.05E-13	3.64E-5
3	α	0.95	0.96	0.95	0.95	
	NE	223	**86**	218	159	∞
	DTP	5.93E-6	1.39E-5	1.14E-5	1.14E-5	
4	α	0.95	0.95	0.95	0.95	
	NE	361	**227**	350	246	∞
	DTP	1.53E-5	1.53E-5	2.12E-5	2.21E-5	
5	α	0.95	0.95	0.95	0.95	
	NE	890	709	883	**703**	∞
	DTP	1.58E-5	3.02E-5	1.64E-5	2.64E-5	
6	α	0.95	0.95	0.95	0.95	
	NE	850	**650**	4214	4214	∞
	DTP	1.97E-5	1.97E-5	2.8E-5	2.79E-5	
7	α	0.95	0.95	0.95	0.95	
	NE	**2316**	**2316**	5987	5708	∞
	DTP	2.86E-5	2.86E-5	2.97E-5	2.59E-5	
8	α	0.95	0.95	0.95	0.95	
	NE	**6090**	**6090**	7445	7445	∞
	DTP	8.38E-5	8.38E-5	7.82E-4	7.82E-4	

Table 3.5 presents the experimental results obtained by the methods. Observe that the SM did not converge for $d \geq 3$, for those number of function evaluations (NE) indicated by the symbol ∞. According to Table 3.5, the LCNM-G offered the best performance. However, the LCNM method tied the LCNM-G method in the cases $d = 7$ and $d = 8$.

Experiment 4b

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$$

subject to C_1, C_2 and C_3 from (3.43).

Here, we used the same initial point and stopping parameters as experiment 4a. However, the experimentation was done for the cases of dimension more than or equal to four, because there exist four constrained variables.

In this case the optimum point is given by $\mathbf{x}_{\min}^T = (50,\ -15,\ 22.5,\ 22.5,\ \underbrace{0,\ldots,0}_{(d-4)\text{ times}}\ )$

Table 3.6: Summary of Experiment 4b.

d		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
	α	0.95	0.93	0.95	0.95	0.85
4	NE	424	295	225	**217**	389
	DTP	1.1E-5	1.59E-5	1.87E-5	1.87E-5	2.59E-5
	α	0.95	0.95	0.95	0.95	
5	NE	1019	878	687	**572**	∞
	DTP	2.46E-5	2.46E-5	2.32E-5	2.32E-5	
	α	0.95	0.95	0.95	0.95	
6	NE	677	**488**	2124	1909	∞
	DTP	1.74E-5	2.56E-5	1.8E-5	1.8E-5	
	α	0.96	0.96	0.96	0.96	
7	NE	2111	**1137**	1192	1192	∞
	DTP	7.58E-5	4E-1	3.86E-1	3.86E-1	
	α	0.94		0.96		
8	NE	16532	∞	**4419**	∞	∞
	DTP	7.14E-5		2.21E-5		

Table 3.6 shows that the LCNM-G method required less number of evaluations than the rest of the methods. However, for the case $d = 8$, the LCNM-G method did not converge. As can be seen from the table, the SM did not work properly, except the case $d = 4$.

Experiment 4c

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$$

subject to C_1, C_2, C_3, C_4 and C_5 from (3.43).

In this case the global optimum point is given by $\mathbf{x}^T_{\min} = (50,\ -15,\ 22.5,\ 22.5,\ 5,\ 5,\ \underbrace{0, \ldots, 0}_{(d-6)\ \text{times}})$ and the initial point is given by $\mathbf{x}^T_{initial} = (100, -100, \underbrace{200, \ldots, 200}_{(d-2)\ \text{times}})$.

Table 3.7: Summary of Experiment 4c.

d		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
	α	0.97	0.97	0.97	0.97	
6	NE	**864**	692	3979	3967	∞
	DTP	1.46E-5	16.154	4.76E-1	4.76E-1	
	α	0.97	0.97	0.97	0.97	
7	NE	3225	**1305**	2889	2310	∞
	DTP	6.42E-5	3.09E-5	1.15E-4	2.33E-5	
	α	0.97	0.97	0.97	0.97	
8	NE	**3631**	**3631**	4804	4663	∞
	DTP	7.88E-1	7.88E-1	1.45E-5	2.52E-5	

According to the results obtained, the best performance was given by the LCNM method for all cases, as is shown in Table 3.7. Observe that the LCNM-G method did not converge to the global optimum point for the case $d = 6$.

Observe that the LCNM method was more robust than its variants and the SM, because the LCNM method satisfactorily identified the TP for all experiments. Nonetheless, the LCNM-G method converged with less number of function evaluations (NE) than the LCNM method in many experiments.

Experiment 5

Comparison using extended Rosenbrock objective function

We also study the performance of the LCNM method and its family versus the SM through a set of experiments, when the objective function to minimize is the so-called extended Rosenbrock function, whose analytical expression is given by

$$f(\mathrm{x}) = \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2 \right]$$

where d is an even integer number.

Here we define a set of C_i linear constraints that will be employed in our constrained optimization test problems

$$\begin{array}{lrcl} C_1: & 3x_1 + 2x_2 & \geq & 12 \\ C_2: & x_1 + 2x_2 & \leq & 2 \\ C_3: & x_1 + x_2 + x_3 + x_4 & \geq & 8 \\ C_4: & x_1 + x_2 + x_3 + x_4 + 0.5x_5 + 0.5x_6 & \geq & 10 \end{array} \qquad (3.44)$$

The initial point was $\mathbf{x}_{\text{min}}^T = (50, \, -50, \underbrace{50, \ldots, 50}_{(d-2) \text{ times}})$ and the parameters of stopping rule $\eta_1 = \eta_2 = 10^{-6}$.

Note that the constraints C_1 and C_2 form a sharp angle, which can produce a large number of function evaluations (NE) for 4 and 6 dimensional problems. Furthermore, the constraints define several sub-spaces. The identification of optimum in this class of problem can require the search of optimum in the sub-spaces.

The optimum point for each case is given by

$$\mathbf{x}_{\text{min}}^T = \begin{cases} (5, \; -1.5) & \text{Exp. 5a} \\ (5, \; -1.5, \; 1.67909, \; 2.82091) & \text{Exp. 5b} \\ (5, \; -1.5, \; -2.67751, \; 7.17751, \; 1.56122, \; 2.43878) & \text{Exp. 5c} \end{cases}$$

Table 3.8: Summary of Experiment 5.

Exp	d	Subject to		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
5a	2	C_1, C_2 from (3.44)	α	0.95	0.95	0.95	0.95	0.90
			NE	16	12	17	13	286
			DTP	2.8E-14	1.97E-14	4.97E-14	3.81E-14	1.74E-7
5b	4	C_1, C_2, C_3 from (3.44)	α	0.95	0.95	0.95	0.95	0.95
			NE	236	128	230	222	4429
			DTP	3.07E-6	3.07E-6	7.17E-7	7.17E-7	99.776
5c	6	C_1, C_2, C_3, C_4 from (3.44)	α	0.95	0.95	0.95	0.95	0.95
			NE	11465	292	14569	318	∞
			DTP	5.47E-6	214.998	97.194	214.525	

According to the results presented in Table 3.8, the best performance was carried out by the LCNM method, because it shows the best accurate in comparison to others within a minor NE for all experiments. Moreover, the LCNM method was the unique method that identified the minimum point for Experiment 5c.

Experiment 6

Comparison using extended Rosenbrock objective function

In this experiment we shall minimize the Rosenbrock function again. However, the feasible region is given by another set of C_i linear constraints, namely

$$\begin{aligned} C_1: &\quad x_1 + x_2 \geq 10 \\ C_2: &\quad -2x_1 + x_2 \leq 20 \\ C_3: &\quad -4x_1 + x_2 \leq 0 \\ C_4: &\quad x_3 + x_4 \geq 12 \end{aligned} \tag{3.45}$$

The parameters of stopping rule were fitted at $\eta_1 = \eta_2 = 10^{-6}$, and we took as initial point for each test $\mathbf{x}_{initial} = 20 \cdot \mathbf{1}_d$.

In this experiment optimum points are:

$$\mathbf{x}_{\min}^T = \begin{cases} (4,16) & \text{Exp. 6a} \\ (4,16,2,9) & \text{Exp. 6b} \\ (4,16,2,9,1,1) & \text{Exp. 6c} \end{cases}$$

Table 3.9: Summary of Experiment 6.

Exp	d	Subject to		LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
6a	2	C_1, C_2, C_3 from (3.45)	α	0.95	0.95	0.95	0.95	0.95
			NE	165	40	125	73	176
			DTP	7.73E-3	13.27	7.74E-3	7.74E-3	16.49
6b	4	C_1, C_2, C_3, C_4 from (3.45)	α	0.95	0.95	0.95	0.95	0.8
			NE	494	432	697	635	919
			DTP	9.9E-1	9.9E-1	9.9E-1	9.9E-1	6.38
6c	6	C_1, C_2, C_3, C_4 from (3.45)	α	0.9	0.95	0.94	0.95	∞
			NE	3337	1346	4151	3078	
			DTP	9.9E-1	41.04	9.9E-1	61.38	

Table 3.9 displays the summary of the experimental results where is shown that the LCNM method had the best performance. Observe that the LCNM-G method did not identify the optimum point for the experiments 6a and 6c.

Experiment 7

Comparison using flat objective function near to the optimum point.

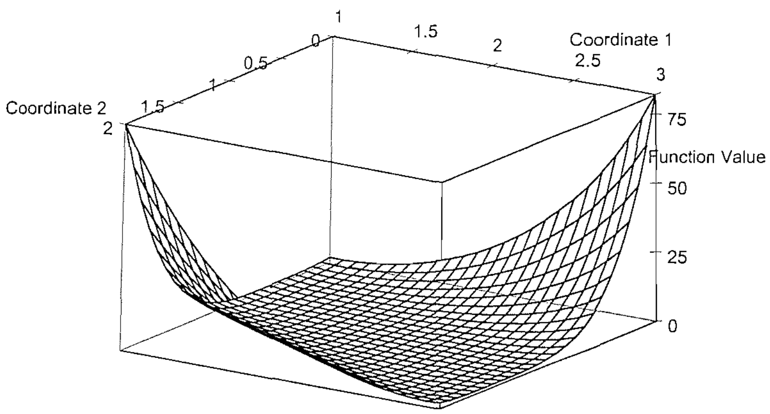


Figure 3.3: Flat function near to optimum.

Figure 3.3 depicts a flat function close to its global optimum point. This kind of optimization problem has a particular difficulty of locating the optimum point on the flat zone of the function. A mathematical formulation of our numerical example is as follows,

$$\min_{\mathbf{x}\in\mathbb{R}^2} \left[(x_1 - 2)^8 + (x_1 - 2x_2)^4\right]$$

subject to:

$$-5x_1 + 10x_2 \geq c$$

where the parameter $c \geq 0$ takes the value of 0 and 0.5

The local minimum of the optimization problem is $\mathbf{x}_{\min} = (2,\ 1 + c/10)^T$. We took as the initial point for all methods the point $\mathbf{x}_{initial} = (200, 200)^T$ and the parameters of stopping rule were fitted at $\eta_1 = \eta_2 = 10^{-6}$.

Table 3.10: Summary of Experiment 7.

c		TP	LCNM	LCNM-G	LCNM-R	LCNM-RG	SM
	α		0.95	0.95	0.95	0.95	0.95
	NE		76	71	100	95	361
0	DTP		4.573E-4	4.573E-4	4.448E-7	4.448E-7	1.415E-6
	$f(x_{\min})$	0.00	7.833E-52	7.833E-52	6.279E-52	6.282E-52	1.415E-46
	x_1	2.00	1.9999996	1.9999996	1.9999996	1.9999996	2.0000013
	x_2	1.00	0.9999998	0.9999998	0.9999998	0.9999998	1.0000006
	α		0.95	0.95	0.95	0.95	0.95
	NE		85	81	∞	∞	643
0.5	DTP		7.646E-3	7.646E-3			1.59E-2
	$f(x_{\min})$	1E-4	0.9999E-4	0.9999E-4			0.9999E-4
	x_1	2.00	1.9931616	1.9931616			1.9857781
	x_2	1.05	1.0465808	1.0465808			1.0428890

From Table 3.10 we note that all methods approach the optimum point satisfactorily for the case $c = 0$. However, the family of the LCNM methods required less NE than the SM when $c = 0$, especially the LCNM-G method that found the optimum point with the lowest NE. Nevertheless, the SM identified the TP with the best accuracy.

On the other hand, the LCNM-R method and the LCNM-RG method were unable to converge to the TP in the case when $c = 0.5$, whilst the rest of the methods reached the optimum point with an exactitude of order 10^{-2}.

3.8 Further comparisons

In this section we shall compare the LCNM method, the Lucidi-Sciandrone-Tseng (LST) method and the Solver of Microsoft® Excel spreadsheet for a group of test problems reported by Lucidi et al. (2002), which are described in (Hock and Schittkowski 1981) and (Schittkowski 1987).

The setting of the LCNM method were fixed as follows: $\alpha = 0.95$, $\beta = 0.5$, $\gamma = 2$, $\delta = 0.5$, $\tau = 1$, $\mu = 1$ and $\rho = 0.99$. With respect to the tolerance errors η was fixed for obtaining a satisfactory value of $\Delta f/f^*$, which is defined by Lucidi et al. (2002) as

$$\frac{\Delta f}{f^*} = \frac{f_m - f^*}{f^*}$$

where f_m is the experimental value which is obtained by the method, and f^* is the minimum value reported by Hock and Schittkowski (1981) and Schittkowski (1987) for each particular problem.

Table 3.11: Comparison of the LCNM method, the LST method and the Solver of Excel

			NE ($\Delta f/f^*$)		
Problem	d	f^*	LCNM	LST	Excel
HS(21)	2	-99.96	21(-10^{-4})	18(0)	(0)
HS(24)	2	-1	22(-10^{-15})	14(0)	(0)
HS(232)	2	-1	17 (-10^{-15})	29 (0)	(0)
HS(331)	2	4.258	61 (10^{-7})	104 (10^{-4})	(10^{-4})
HS(36)	3	-3300	36(10^{16})†	24(0)	(0)
HS(37)	3	-3456	112(-10^{-13})	113(10^{-11})	(0)
HS(251)	3	-3456	112 (-10^{-13})	153 (10^{-11})	(0)
HS(340)	3	-0.054	99 (-10^{-12})	108 (10^{-8})	(0)
HS(44)	4	-15	116(10^{-14})‡	22(0)	(0)
HS(76)	4	-4.681818181	106(10^{-10})	103(10^{-9})	(10^{-10})
HS(354)	4	0.113784	190 (-10^{-4})	236 (10^{-4})	(-10^{-6})

†: $\alpha = 1$ ‡: $\mathbf{x}_{ini}=0.1\cdot\mathbf{1}_4$

We selected the best reported result of both the LCNM method and the Lucidi-Sciandrone-Tseng (LST) method, within an adequate exactitude.

Although, this group of test problems is a non-definitive proof of the potentiality of the LCNM method in comparison with the LST method and the Solver of Microsoft® Excel spreadsheet, it may be appreciated from Table 3.11, the satisfactory accuracy of the LCNM method within a smaller NE in 7 of the 11 test problems performed herein.

3.9 Computational effort

In this section, we shall study again through examples the computational effort required by the LCNM method for finding the optimum solution. We consider especially the NE. This performance measure of the algorithm was taking into account because the NE has an important role when this kind of method is applied in the optimization of complex system, such as, the identification of optimum operation in simulation models of stochastic systems. In these experiments, the step size parameter τ of the LCNM algorithm was fixed to 0.2 and the parameters $\alpha = 0.95$, $\beta = 0.5$, $\gamma = 2$ and $\delta = 0.5$.

3.9.1 Experiment 1: Convex quadratic objective function subject to a linear constraint

In this computational effort experiment, we shall study the NE and its DTP for different values of $\eta_1 = \eta_2 = \eta$ and dimension d.

$$\min_{\mathbf{x} \in \mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $\sum_{i=1}^{d} x_i \geq 10d$.

The local minimum of the optimization problem is $\mathbf{x}_{\text{min}} = 10 \cdot \mathbf{1}_d$ and as initial point $\mathbf{x}_{initial} = 100 \cdot \mathbf{1}_d$.

Table 3.12: Summary of Computational Effort in Experiment 1.

		η		
		10^{-2}	10^{-4}	10^{-6}
d	1	10 1.77E-15	10 1.77E-15	10 1.77E-15
	2	52 7.33E-3	78 8.22E-3	102 5.84E-6
	3	91 5.08E-2	134 5.33E-4	172 5.91E-6
	4	217 6.97E-2	365 6.83E-4	498 7.80E-6
	5	288 8.55E-2	548 1.07E-3	714 7.68E-6
	6	482 9.95E-2	682 9.78E-4	1144 9.38E-6
	7	778 8.23E-1	1074 7.23E-1	1833 4.63E-2
	8	659 4.10E-1	983 3.27E-3	1159 1.48E-5

Table 3.12 depicts the NE and the DTP for each dimension from 1 to 8 and three values of η. Note that for the case of dimension 7 and 8, the NE and the DTP produced a response different to the rest of the cases.

3.9.2 Experiment 2: Quadratic objective function subject to two linear constraints

Here, we shall consider the case of a quadratic objective function subject to two linear constraints for estimating the computational effort of the algorithm for three cases of stopping rule tolerance $\eta_1 = \eta_2 = \eta$, and dimension cases from 2 to 6. The optimization problem studied is

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $3x_1 + 2x_2 \geq 120$ and $x_1 + 2x_2 \leq 20$.

For this computational effort test, the local optimum solution is given by

$\mathbf{x}_{\min} = (50, -15, \underbrace{0, 0, \ldots, 0}_{(d-2)\text{ times}})^T$ with $f(\mathbf{x}_{\min}) = 2725$, and we considered as initial point for building the simplex the point $\mathbf{x}_{initial} = (400,\ -400, \underbrace{400, \ldots, 400}_{(d-2)\text{ times}})$.

As a result of the experimentation, we obtained the following table.

Table 3.13: Summary of Computational Effort in Experiment 2.

		η		
		10^{-2}	10^{-4}	10^{-6}
d	2	22 7.69E-14	22 7.69E-14	22 7.69E-14
	3	120 4.89E-2	175 1.38E-3	223 5.93E-6
	4	214 2.06E-1	286 1.11E-4	361 1.53E-5
	5	201 76.808	773 1.39E-3	890 1.58E-5
	6	572 2.19E-1	721 2.35E-3	850 1.97E-5

Table 3.13 displays NE and DTP for each dimension from 2 to 6 and three values of η. As is shown in Table 3.13, the NE increased when the dimension of the mathematical model was increased for a given tolerance η. Furthermore, according to the figures of the table, the NE increased as result of a reduction of the tolerance η, for the cases of dimensions from 3 to 6.

3.10 Conclusions

The LCNM method has been shown to have a valuable potential for the identification of optima of linearly constrained non-linear optimization problems, as a result of reducing the number of vertices of the simplex, and thereby reducing the number of evaluations of the objective function. Furthermore, the LCNM method has shown its advantage in comparison to the method of Subrahmanyam, for the case of non-linear optimization problems subject to linear constraints, without detracting from the importance of the point of view of the method of Subrahmanyam, which is an approach to non-linear optimization problems subject to both linear constraints and non-linear constraints.

Moreover, according to the comparative study presented in Section 3.7, the LCNM method was more robust than the other variant forms, because the LCNM method satisfactorily identified the optimal solution for all experiments, whereas variants failed in some cases. According to numerical results reported in Section 3.7, the LCNM-G was shown to be slightly more economic than the LCNM method. This fact can be easy explained, because the LCNM method computes the gradient of the objective function for choosing the active constraints in accordance with Criterion 3.1 (page 27).

In addition, the LCNM method has also been shown to be a competitive method in comparison to the LST method, due to its exactitude within a minor number of function evaluations in 7 of 11 test problems carried out in Section 3.8. Nevertheless, this fact, it not should be considered as a definite evidence of the superiority of the LCNM method.

Chapter 4

Properties of the LCNM method

4.1 Introduction

Ideally the study of convergence of any optimization algorithm requires a deep analysis of the algorithm under general conditions. In this chapter the LCNM algorithm has been studied through case studies and some general features identified when the objective function is strictly convex in a polyhedral feasible region.

This chapter deals with some general conditions under which the LCNM method converges. However, recently McKinnon (1998) shows how the NM method can converge to a non-stationary point, when it is applied for minimizing a class of 2-dimensional strictly convex objective functions, causing repeated focused inside contraction (RFIC) operations, so inside contraction steps are repeatedly attempted by the NM method, leaving fixed the best vertex of the simplex, when this vertex could be a non-stationary point. This fact could weaken our study of convergence presented in this chapter.

Nonetheless Kelley (1999) proposes a test for sufficient decrease, where the rate of decreasing of the average objective function value is measured for detecting the convergence to non-stationary points of unconstrained minimization problems. If this rate is not held, the modified NM method of Kelley restarts the simplex to a smaller one with orthogonal edges. The procedure of Kelley (1999) was not included in the LCNM method. This is because though Kelley guarantees convergence of the NM algorithm to a stationary point, if the objective function is smooth, the procedure can cause needless reinitializations of the simplex, if the objective function is non-smooth or if it is corrupted by noise.

Furthermore, Price et al. (2002) present another perspective for ensuring the convergence of the NM algorithm to a non-stationary point, whereby, if an iteration of the NM algorithm does not produce a good enough descent, the method redefines a new simplex based on a grid. The proposal of Price and colleagues looks like a very interesting method, because numerical tests verify its convergence to a stationary point, even in the class of functions studied by McKinnon (1998). However, according to a set of 39 numerical test problems carried out by Price et al. (2002), we observe that in 22 of their numerical tests, the modified NM algorithm was more expensive, with respect to the number of function evaluations, than the original NM simplex algorithm. This fact persuaded us not to consider it as a possible approach in our development of the LCNM algorithm.

Nonetheless, numerical examples have illustrated some advantages of the LCNM method, even in situations when the objective function contains noise. This last aspect is one of the main aims of this research, because our interest is focused on the developing of a constrained optimization method for noisy objective functions.

This chapter is organized as follows: In Section 4.2 we shall study some properties of the method when no local minimum exists at any internal point of the feasible region. A study of convergence for a particular cases of a triangular simplex is shown in Section 4.3. In addition, a rigorous study of convergence for the case of linear objective function subject to two linear symmetric constraints is presented in Section 4.4. Finally conclusions of the considered approach on this chapter are reported in Section 4.5.

4.2 Search properties of the method

In this section, we shall show some search properties of the method for the case when there exists at least a local minimum on the linear constraint boundary. We begin by formulating a basic representation of a typical simplex iteration. This will be used to study the performance of the method. We shall in particular use it to show general properties of the method for strictly convex objective function subject to k linear inequality constraints.

Here we shall develop a theoretical framework for explaining how an initial simplex is moved by the LCNM algorithm, when there exists no local minimum at any interior point of the feasible region. The simplex therefore approaches the boundary of the feasible region. We shall also study the transformation of the simplex and its movement in order to identify a sub-space that contains at least a local minimum.

4.2.1 Transformation of the simplex

To describe the transformation that is carried out after any simplex operation and when a new active constraint is obtained, we shall first consider how to represent the $(q+1)$th simplex from a qth iteration. We should remark that the LCNM algorithm is made up of s stages, each stage comprising a number of iterations. Each one of the iterations is principally defined by the sort of vertices encountered in the current qth simplex, by the basic simplex operations and by a step for verifying if a new linear constraint was activated by the current qth simplex. We hereinafter suppress the stage counter sth for simplifying the notation in the study of search properties of the LCNM algorithm.

Let $S_{v_q}^{[q]}$ denote a matrix at the qth iteration, whose columns represent the v_q d-dimensional $\mathbf{x}_i^q$ vertices, where they are arranged in ascending order of the objective function value $f(\mathbf{x}_i^q)$, and its vertices have activated r_q $(0 \leq r_q \leq k)$ linear constraints in terms of the Definition 3.7. Thus $S_{v_q}^{[q]}$ represents a $d \times v_q$ matrix, where $v_q = d + 1 - r_q$. In consequence, $S_{v_q}^{[q]}$ is represented as follows

$$S_{v_q}^{[q]} = \left[\mathbf{x}_1^q : \mathbf{x}_2^q : \ldots : \mathbf{x}_{v_q-1}^q : \mathbf{x}_{v_q}^q\right] \tag{4.1}$$

where the subscript of each column vector means the ranking of the vertices according to the value of the objective function at each vertex. In other words, $\mathbf{x}_1^q$ is the vertex whose function value $f(\mathbf{x}_1^q)$ is less than or equal to $f(\mathbf{x}_2^q)$, $\mathbf{x}_2^q$ is the vertex whose function value $f(\mathbf{x}_2^q)$ is less than or equal to $f(\mathbf{x}_3^q)$ and so forth.

Let $E_p^{[q]}$ represent the $d \times (v-1)$ matrix whose jth column means the edge of $S_v^{[q]}$ between $\mathbf{x}_p^q$ and $\mathbf{x}_j^q$ for all $p \neq j = 1, 2, \ldots, v_q$, thus

$$E_p^{[q]} = \left[\mathbf{x}_1^q - \mathbf{x}_p^q : \mathbf{x}_2^q - \mathbf{x}_p^q : \cdots : \mathbf{x}_{j \neq p}^q - \mathbf{x}_p^q : \cdots : \mathbf{x}_{v_q}^q - \mathbf{x}_p^q\right] \; \forall p = 1, \ldots, v \tag{4.2}$$

We define the diameter of $S_{v_q}^{[q]}$ as the maximal distance between any two vertices of the current simplex $S_{v_q}^{[q]}$, thus

$$diam\left(S_{v_q}^{[q]}\right) = \max_{i \neq j} \left\|\mathbf{x}_i^q - \mathbf{x}_j^q\right\| \tag{4.3}$$

A criterion of stopping of the method within any sth stage is based on this quantity, whose value is verified after each qth iteration of the algorithm.

Using the same approach of Lagarias et al. (1998), we shall study the transformation of $S_{v_q}^{[q]}$ into $S_{v_q}^{[q+1]}$ due to non-shrink and shrink operations.

A non-shrink operations can be generically represented by

$$\mathbf{x}^q_{new} = (1+\theta)\mathbf{x}^q_{cen} - \theta\mathbf{x}^q_{v_q} \tag{4.4}$$

where the parameter θ is given by

$$\theta = \begin{cases} \alpha & \text{Reflection operation} \\ \alpha\gamma & \text{Expansion operation} \\ -\beta & \text{Inside contraction operation} \end{cases} \tag{4.5}$$

Due to Proposition 3.3 and Proposition 3.4, the coordinates of $\mathbf{x}^q_{new}$ are modified by the Linear Constraint Procedure (LCP) after an operation of reflection or expansion, if $\mathbf{x}^q_{new}$ does not belong to the feasible region. In this case, θ is fitted to λ by the LCP and so, Equation (4.5) can be rewritten as

$$\theta = \begin{cases} \min(\alpha,\lambda) & \text{Constrained reflection operation} \\ \min(\alpha\gamma,\lambda) & \text{Constrained expansion operation} \\ -\beta & \text{Inside contraction operation} \end{cases} \tag{4.6}$$

where $\lambda = \min(\lambda_1,\ldots,\lambda_k)$, $\lambda_i \in \{\mathbb{R} \mid \lambda_i \geq 0 \ \forall i = 1,\ldots k\}$ is given by Equation (4.7) for each ith linear constraint of the feasible region defined by the k linear constraints.

$$\lambda_i = \frac{b_i - a_i^T\mathbf{x}^q_{cen}}{a_i^T\left[\mathbf{x}^q_{cen} - \mathbf{x}^q_{v_q}\right]}, \qquad \forall i = 1,\ldots,k \tag{4.7}$$

Note that λ_i may be negative, in this case it is not considered for calculating λ, because the intersection point on the ith linear constraint boundary is in the direction from $\mathbf{x}_{cen}$ towards $\mathbf{x}_{\max}$, whose direction is opposite to the descent directional vector $\mathbf{d} = -\nabla f(\mathbf{x})$.

Using Equation (3.13) in Equation (4.4), we have that the trial point $\mathbf{x}^q_{new}$ is given by

$$\mathbf{x}^q_{new} = \frac{(1+\theta)}{v_q - 1}\left[\mathbf{x}^q_1 + \mathbf{x}^q_2 + \ldots + \mathbf{x}^q_{v_q-1}\right] - \theta\mathbf{x}^q_{v_q} \tag{4.8}$$

Using matrix notation, Equation (4.8) becomes

$$\mathbf{x}^q_{new} = S^{[q]}_{v_q}\ \mathbf{t}(\theta, v_q) \tag{4.9}$$

where $\mathbf{t}(\theta,v_q) = \left[\frac{(1+\theta)}{v_q-1}, \frac{(1+\theta)}{v_q-1}, \ldots, \frac{(1+\theta)}{v_q-1}, -\theta\right]^T$ is a column vector of dimension $v_q \times 1$.

If one of these operations is attempted during a step of reflection, expansion or contraction, the maximum vertex $\mathbf{x}^q_{v_q}$ will be substituted by the trial point $\mathbf{x}^q_{new}$ and, this latter is converted into a vertex of the $(q+1)$th simplex. Hence a not necessarily unorganized simplex $\tilde{S}^{[q+1]}_{r_q}$ at the beginning of the $(q+1)$th iteration will be represented by

$$\tilde{S}^{[q+1]}_{v_{q+1}} = \left[\mathbf{x}^q_1 : \mathbf{x}^q_2 : \ldots : \mathbf{x}^q_{v_q-1} : \mathbf{x}^q_{new}\right] \tag{4.10}$$

Due to this replacement, the simplex might therefore require an additional adjustment, depending on whether it activates a new linear constraint or not. This adjustment simply reduces the number of vertices v_q if a constraint were activated. More precisely, the algorithm would choose the $(v_q - 1)$ best vertices from the current simplex because of Criterion 3.2. Observe that this latter transformation is carried out in an operation of constrained reflection or constrained expansion only. It cannot occur in contraction operations because the LCNM algorithm only uses the so-called inside contraction Nelder-Mead operations.

For a shrink operation, the simplex $S^{[q]}_{v_q}$ is transformed into

$$\tilde{S}^{[q+1]}_{v_{q+1}} = \left[(1-\delta)\mathbf{x}^q_1 + \delta\mathbf{x}^q_1 : (1-\delta)\mathbf{x}^q_1 + \delta\mathbf{x}^q_2 : \ldots : (1-\delta)\mathbf{x}^q_1 + \delta\mathbf{x}^q_{v-1} : (1-\delta)\mathbf{x}^q_1 + \delta\mathbf{x}^q_{v_q}\right] \tag{4.11}$$

By rearranging the terms, Equation (4.11) becomes

$$\tilde{S}^{[q+1]}_{v_{q+1}} = (1-\delta)\mathbf{x}^q_1\mathbf{1}^T_v + \delta S^{[q]}_{v_q} \tag{4.12}$$

Using Equation (4.10) and Equation (4.12), the transformed $(q+1)$th simplex is defined by

$$\tilde{S}^{[q+1]}_{v_{q+1}} = \begin{cases} \underbrace{\left[\mathbf{x}^q_1 : \mathbf{x}^q_2 : \ldots : \mathbf{x}^q_{v_q-1} : \mathbf{x}^q_{new}\right]}_{v_q \text{ vertices}} & \text{Constrained non-shrink simplex operation, due to no new active constraint, where } v_{q+1} = v_q, \\ \underbrace{\left[\mathbf{x}^q_1 : \mathbf{x}^q_2 : \ldots : \mathbf{x}^q_{v_q-1} : \mathbf{x}^q_{new}\right]}_{v_q-1 \text{ best vertices}} & \text{Constrained non-shrink simplex operation, due to a new active constraint, where } v_{q+1} = v_q - 1 \\ (1-\delta)\mathbf{x}^q_1 \mathbf{1}^T_{v_q} + \delta S^{[q]}_{v_q} & \text{Shrink Nelder-Mead operation where } v_{q+1} = v_q. \end{cases} \tag{4.13}$$

Since a new linear constraint can be only activated by either a constrained reflection operation or a constrained expansion operation to a simplex $S^{[q]}_{v_q}$, these operations will be generally named *constrained simplex operations* hereinafter.

4.2.2 General properties of the Linear Constrained Nelder-Mead algorithm

Here we present some general properties of the LCNM algorithm, considering the basic properties of the NM method established by Lagarias et al. (1998), who studied the properties of the NM method from the perspective of the d-dimensional volume of the simplex, whose value is computed through the determinant of the matrix $E^{[q]}_p$ given by Equation (4.2). Some basic properties can be extended to our case, because the LCNM method always fits the simplex to the r_q active linear constraints by reducing the number of vertices.

Lemma 4.1 (Strictly convexity of a function) *Let $A = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \subset \Omega$ be a subset of points, where Ω is a non-empty convex set of $\mathbb{R}^d$. If $f(\mathbf{x})$ is a strictly convex function on the region Ω, for any integer $n \geq 2$, then*

$$f\left(\sum_{i=1}^{n} \mu_i \mathbf{x}_i\right) < \sum_{i=1}^{n} \mu_i \, f(\mathbf{x}_i) \tag{4.14}$$

where $\sum_{i=1}^{n} \mu_i = 1$, $\mu_i > 0$.

Moreover

$$f\left(\sum_{i=1}^{n} \mu_i \mathrm{x}_i\right) < \max\{f(\mathrm{x}_1), f(\mathrm{x}_2), \ldots, f(\mathrm{x}_n)\} \tag{4.15}$$

Proof. See Appendix B. ■

We must remark that the subscripts of x_i in the set A of the above lemma does not require an ordered sequence of x_i's.

Lemma 4.2 (Convexity of a linear subset) *Let $\mathcal{F}$ denote a non-empty set of feasible points in $\mathbb{R}^d$. If $\mathcal{F}$ is defined by a set of linear inequality constraints $\mathcal{F} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A\mathrm{x} \geq \mathbf{b}\}$, where matrix A and vector $\mathbf{b}$ are consistent, then $\mathcal{F}$ is a convex set of $\mathbb{R}^d$.*

Proof. See Appendix B. ■

Lemma 4.3 (Relative value of $f(\mathrm{x}^q_{cen})$) *Consider the problem $\mathcal{P}$ of minimizing an objective function $f(\mathrm{x})$ on $\mathbb{R}^d$ using the LCNM method. Let x^q_{cen} be the centroid of the remaining hyperface $H^{[q]}_{v_q}$ of the current simplex $S^{[q]}_{v_q}$, which is contained on a non-empty convex set Ω. If the objective function $f(\mathrm{x})$ is strictly convex on the set Ω, then $f(\mathrm{x}^q_{cen}) < f(\mathrm{x}^q_{v_q-1}) \leq f(\mathrm{x}^q_{v_q})$.*

Proof. Let $S^{[q]}_{v_q} = \left[\mathrm{x}^q_1 : \mathrm{x}^q_2 : \ldots : \mathrm{x}^q_{v-1} : \mathrm{x}^q_{v_q}\right]$ be the set of sorted vertices that defines our current simplex, such that $f(\mathrm{x}^q_1) \leq f(\mathrm{x}^q_2) \leq, \ldots, \leq f(\mathrm{x}^q_{v_q-1}) \leq f(\mathrm{x}^q_{v_q})$. Because of the LCNM method, from Equation (3.13) and (4.14) we have that if

$$\mathrm{x}^q_{cen} = \frac{1}{v_q - 1} \sum_{i=1}^{v_q-1} \mathrm{x}^q_i \;\text{ with }\; 0 < \frac{1}{v_q - 1} < 1 \;\text{ and }\; \sum_{i=1}^{v_q-1} \frac{1}{v_q - 1} = 1, \;\; \forall v_q \geq 2 \tag{4.16}$$

then by Lemma 4.1, we obtain

$$\begin{gathered} f(\mathrm{x}^q_{cen}) < \textstyle\sum_{i=1}^{v_q-1} \frac{1}{v_q-1} \, f(\mathrm{x}^q_i) \;\text{ and hence} \\ f(\mathrm{x}^q_{cen}) < \max\left\{f(\mathrm{x}^q_1), f(\mathrm{x}^q_2), \ldots, f(\mathrm{x}^q_{v_q-1})\right\} \end{gathered} \tag{4.17}$$

where $f(\mathrm{x}^q_{v_q-1}) = \max\left\{f(\mathrm{x}^q_1), f(\mathrm{x}^q_2), \ldots, f(\mathrm{x}^q_{v_q-1})\right\}$, because $S^{[q]}_{v_q}$ is defined by an ordered sequence of vertices x^q_i. Therefore $f(\mathrm{x}^q_{cen}) < f(\mathrm{x}^q_{v_q-1}) \leq f(\mathrm{x}^q_{v_q})$. ■

Lemma 4.4 (Improving of x^q_{new}) *Let $\mathcal{F}$ be a non-empty feasible convex set of $\mathbb{R}^d$ such that $\mathcal{F} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A\mathrm{x} \geq \mathbf{b}\}$, where matrix A and vector $\mathbf{b}$ are consistent, and let x^q_{new}*

represent a feasible trial point in $\mathcal{F}$, which is yielded by either a constrained reflection operation with $\theta = \min(\alpha, \lambda)$ or a constrained expansion operation with $\theta = \min(\alpha\gamma, \lambda)$ on a current simplex $S_{v_q}^{[q]} = [\mathbf{x}_1^q : \mathbf{x}_2^q : \ldots : \mathbf{x}_{v-1}^q : \mathbf{x}_{v_q}^q]$. If $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ is a strictly convex and differentiable function on a non-empty convex feasible set $\mathcal{F}$ and with $\nabla f(\mathbf{x}_{new}^q)^T [\mathbf{x}_{cen}^q - \mathbf{x}_{v_q}^q] < 0$ for all $0 \leq \theta \leq \min(\alpha, \lambda)$ or $0 \leq \theta \leq \min(\alpha\gamma, \lambda)$ according to the kind of constrained operation, and $\nabla f(\mathbf{x}_{int}) \neq \mathbf{0}_d$ for all $\mathbf{x}_{int}$ in the interior of $\mathcal{F}$, then

1) For any value of θ defined above, $f(\mathbf{x}_{new}^q) < f(\mathbf{x}_{v_q-1}^q) \leq f(\mathbf{x}_{v_q}^q)$.

2) There exists no local minimum at any $\mathbf{x}_{int}$ in the interior of $\mathcal{F}$.

Proof. PART 1.

For any constrained simplex operation, we have from Equation (4.4)

$$\mathbf{x}_{new}^q = (1+\theta)\mathbf{x}_{cen}^q - \theta \mathbf{x}_{v_q}^q$$

where $0 \leq \theta \leq \min(\alpha, \lambda)$ or $0 \leq \theta \leq \min(\alpha\gamma, \lambda)$ depending on the kind of constrained simplex operation involved.

Since $\mathbf{x}_{new}^q$ is a trial point yielded by the projection of $\mathbf{x}_{v_q}^q$ through $\mathbf{x}_{cen}^q$, then we can define a directional vector $\mathbf{d}^q = \mathbf{x}_{cen}^q - \mathbf{x}_{v_q}^q$. Thus, $\mathbf{x}_{new}^q(\theta)$ is defined as

$$\mathbf{x}_{new}^q(\theta) = \mathbf{x}_{cen}^q + \theta \mathbf{d}^q \tag{4.18}$$

By differentiability of $f(\mathbf{x})$ at $\mathbf{x}_{new}^q(\theta)$, we can assure that if $\Delta\theta \to 0$ for $\Delta\theta > 0$, hereinafter, this latter condition will be denoted by $\Delta\theta \downarrow 0$,

$$f[\mathbf{x}_{new}^q(\theta + \Delta\theta)] = f[\mathbf{x}_{new}^q(\theta)] + \Delta\theta \ \nabla f[\mathbf{x}_{new}^q(\theta)]^T \mathbf{d} + \Delta\theta \ \|\mathbf{d}\| \ \psi[\mathbf{x}_{new}^q(\theta) : \Delta\theta \ \mathbf{d}]$$

Rearranging the terms and dividing by $\Delta\theta$, we obtain

$$\frac{f[\mathbf{x}_{new}^q(\theta + \Delta\theta)] - f[\mathbf{x}_{new}^q(\theta)]}{\Delta\theta} = \nabla f[\mathbf{x}_{new}^q(\theta)]^T \mathbf{d} + \|\mathbf{d}\| \ \psi[\mathbf{x}_{new}^q(\theta) : \Delta\theta \ \mathbf{d}]$$

Because $\nabla f[\mathbf{x}_{new}^q(\theta)]^T \mathbf{d} < 0$ for $0 \leq \theta \leq \min(\alpha, \lambda)$ or $0 \leq \theta \leq \min(\alpha\gamma, \lambda)$ depending on the kind of constrained simplex operation, and $\psi[\mathbf{x}_{new}^q(\theta) : \Delta\theta \ \mathbf{d}] \to 0$ as $\Delta\theta \downarrow 0$, we have

$$\frac{f[\mathbf{x}_{new}^q(\theta + \Delta\theta)] - f[\mathbf{x}_{new}^q(\theta)]}{\Delta\theta} < 0$$

Therefore

$$f\left[\mathbf{x}_{new}^{q}(\theta+\Delta\theta)\right] < f\left[\mathbf{x}_{new}^{q}(\theta)\right] \quad \forall\ 0 \leq \theta \leq \min(\alpha,\lambda) \text{ or } 0 \leq \theta \leq \min(\alpha\gamma,\lambda) \tag{4.19}$$

Evaluating Equations (4.18) and (4.19) at $\theta = 0$, and applying Lemma 4.3, we have

$$f\left[\mathbf{x}_{new}^{q}(\Delta\theta)\right] < f\left(\mathbf{x}_{cen}^{q}\right) < f(\mathbf{x}_{v_q-1}^{q}) \leq f(\mathbf{x}_{v_q}^{q}) \text{ as } \Delta\theta \downarrow 0 \tag{4.20}$$

Since Equation (4.19) is satisfied for all $0 \leq \theta \leq \min(\alpha, \lambda)$ or $0 \leq \theta \leq \min(\alpha\gamma, \lambda)$ because of the type of constrained simplex operation, and using Equation (4.20), we obtain

$$f\left[\mathbf{x}_{new}^{q}(\theta)\right] < f\left(\mathbf{x}_{cen}^{q}\right) < f(\mathbf{x}_{v_q-1}^{q}) \leq f(\mathbf{x}_{v_q}^{q}),\quad \forall\ 0 < \theta \leq \min(\alpha,\lambda) \text{ or } 0 < \theta \leq \min(\alpha\gamma,\lambda)$$

Therefore $f(\mathbf{x}_{new}^{q}) < f(\mathbf{x}_{v_q-1}^{q}) \leq f(\mathbf{x}_{v_q}^{q})$.

PART 2.

Since $\nabla f(\mathbf{x}_{int}) \neq \mathbf{0}_d$ for all $\mathbf{x}_{int}$ in the interior of $\mathcal{F}$, we can say that there exists no local minimum in the interior of $\mathcal{F}$, because the first order necessary condition for the existence of a local minimum at $\mathbf{x}^*$ is that $\nabla f(\mathbf{x}^*) = \mathbf{0}_d$ (Bazaraa and Shetty 1979). ■

Theorem 4.1 (Moving into the boundary) *Consider the problem $\mathcal{P}$ of minimizing a strictly convex and differentiable objective function $f(\mathbf{x})$ in $\mathbb{R}^d$ subject to $\mathbf{x} \in \mathcal{F}$, where $\mathcal{F}$ denotes a non-empty set of feasible points given by a set of linear inequality constraints such that $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$. If the LCNM algorithm is applied to the problem $\mathcal{P}$ beginning with an entire initial simplex $S_{v0}^{[0]}$, which is transformed by q LCNM iterations into an entire simplex $S_{v_q}^{[q]}$, and $\nabla f(\mathbf{x}_{int}) \neq \mathbf{0}_d$ for all $\mathbf{x}_{int}$ in the interior of $\mathcal{F}$, then*

1) Each step is either a constrained reflection step or a constrained expansion step at each qth iteration until the collapse of the simplex.

2) The entire simplex $S_{v_q}^{[q]}$ will move into the boundary of $\mathcal{F}$ and one of its vertices reaches at least one of the linear inequality constraint boundaries at the $(q{+}h)$th iteration.

Proof. PART 1. Since the objective function $f(\mathbf{x})$ is strictly convex and differentiable in $\mathcal{F}$, then by Lemma 4.4, we have that each either constrained reflection operation or constrained expansion operation generates a trial point $\mathbf{x}_{new}^{q}$ such that

$$f(\mathbf{x}_{new}^{q}) < f(\mathbf{x}_{v_q-1}^{q}) \leq f(\mathbf{x}_{v_q}^{q}) \tag{4.21}$$

Because the LCNM algorithm attempts either constrained reflection step or constrained expansion step, before seeking its contraction step at qth each iteration, and Equation 4.21 holds true for all $0 < q < h$. The constrained reflection step takes place at the qth iteration if $f(\mathbf{x}_1^q) < f(\mathbf{x}_{new}^q) \le f(\mathbf{x}_{v_q-1}^q)$, otherwise the constrained expansion step is carried out as a result of $f(\mathbf{x}_{new}^q) < f(\mathbf{x}_1^q)$.

PART 2.

To prove this part, we must demonstrate that there exists at least a linear constraint, which is approached by $\mathbf{x}_{new}^q$ where $\mathbf{x}_{new}^q$ is yielded by a kind of constrained simplex operation. From (Gill et al. 1991), we define the residual for an ith linear constraint $\mathbf{a}_i^T\mathbf{x} \ge b_i$ at the $\bar{\mathbf{x}}$ as the scalar $\breve{r}_i(\bar{\mathbf{x}}) = \mathbf{a}_i^T\bar{\mathbf{x}} - b_i$. Note that $\breve{r}_i(\bar{\mathbf{x}})$ is positive when the ith constraint is strictly satisfied at $\bar{\mathbf{x}}$, zero when the point $\bar{\mathbf{x}}$ lies the ith constraint boundary, and negative when the ith constraint is violated by $\bar{\mathbf{x}}$. Using this definition, we say that if the trial point $\mathbf{x}_{new}^q$ approaches to a ith linear constraint boundary, then $\breve{r}_i(\mathbf{x}_{new}^q) < \breve{r}_i(\mathbf{x}_{cen}^q) < \breve{r}_i(\mathbf{x}_{v_q}^q)$.

From Equation (4.4), we have

$$\begin{aligned} \breve{r}_i(\mathbf{x}_{new}^q) &= \mathbf{a}_i^T\mathbf{x}_{new}^q - b_i \\ &= \mathbf{a}_i^T\left[\mathbf{x}_{cen}^q + \theta(\mathbf{x}_{cen}^q - \mathbf{x}_{v_q}^q)\right] - b_i \\ &= \mathbf{a}_i^T\mathbf{x}_{cen}^q - b_i + \theta\ \left[\mathbf{a}_i^T\mathbf{x}_{cen}^q - \mathbf{a}_i^T\mathbf{x}_{v_q}^q\right] \end{aligned} \tag{4.22}$$

Grouping the terms of Equation (4.22) and using the definition of residual, we have

$$\breve{r}_i(\mathbf{x}_{new}^q) = \breve{r}_i(\mathbf{x}_{cen}^q) + \theta\ \mathbf{a}_i^T\left[\mathbf{x}_{cen}^q - \mathbf{x}_{v_q}^q\right]$$

If there exists at least a linear constraint which satisfies

$$\mathbf{a}_i^T\left[\mathbf{x}_{cen}^q - \mathbf{x}_{v_q}^q\right] < 0 \tag{4.23}$$

then

$$\breve{r}_i(\mathbf{x}_{new}^q) - \breve{r}_i(\mathbf{x}_{cen}^q) < 0, \tag{4.24}$$

because the parameter θ is positive when $\mathbf{x}_{new}^q$ and $\mathbf{x}_{cen}^q$ are different.

From Inequality (4.23) we have

$$\begin{aligned} \mathbf{a}_i^T \mathbf{x}_{cen}^q - b_i - \mathbf{a}_i^T \mathbf{x}_{v_q}^q + b_i < 0 \\ \breve{r}_i(\mathbf{x}_{cen}^q) - \breve{r}_i(\mathbf{x}_{v_q}^q) < 0 \end{aligned} \tag{4.25}$$

Using Inequalities (4.24) and (4.25), we obtain

$$\breve{r}_i(\mathbf{x}_{new}^q) < \breve{r}_i(\mathbf{x}_{cen}^q) < \breve{r}_i(\mathbf{x}_{v_q}^q) \tag{4.26}$$

In addition, because Part 1 of this theorem is satisfied for all internal points of $\mathcal{F}$, we can affirm that it exclusively attempts either a constrained reflection step or a constrained extension step at each qth iteration of the LCNM algorithm, when it has not been activated any linear constraint. Therefore, we say that every trial point $\mathbf{x}_{new}^q$ at least approaches to an ith linear constraint boundary for $1 < q < q+h$, where h is the maximum number of required iterations for reaching at least an ith linear constraint. ∎

Theorem 4.1 would explain us the reason why a linear constraint can be become active when an initial entire simplex $S_{v0}^{[0]}$ is transformed by the LCNM method and there exists no local minimum inside the feasible region $\mathcal{F}$, whereby the number of vertices of the current simplex is reduced by the LCNM method.

Furthermore, Theorem 3.5 (on page 40) explains us how a simplex, whose vertices are on a linear constraint boundary of the feasible region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$ except the vertex $\mathbf{x}_{v_q}^q$, is flattened by a constrained simplex operation.

We shall introduce further lemmas and theorems for showing how the LCNM algorithm identifies an intersection feasible region, which can contain at least a local minimum.

Lemma 4.5 (Intersection of two convex sets) *Let D_1 and D_2 be non-empty convex sets in $\mathbb{R}^d$. If $D_1 \cap D_2$ is a non-empty subset of both D_1 and D_2, then $D_1 \cap D_2$ is a non-empty convex subset in $\mathbb{R}^d$.*

Proof. See Appendix B. ∎

Theorem 4.2 (Convex function on a linear constraint) *Let $\mathcal{L}$ be a non-empty convex set of $\mathbb{R}^d$ defined by a linear equality $\mathcal{L} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid a^T\mathbf{x} = b\}$, where $a^T \in \mathbb{R}^d$ is a vector of constants and b is a real scalar. If $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ is a convex function on a non-empty convex set Ω of $\mathbb{R}^d$, and $\mathcal{L} \cap \Omega$ is a non-empty convex subset of Ω, then $f(\mathbf{x})$ is a convex function on the subset $\mathcal{L} \cap \Omega$.*

Proof. See Appendix B. ∎

Note that according to Theorem 4.2, if x_1 and x_2 belong to $\mathcal{L} \cap \Omega$, then they also belong to $\mathcal{L} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid a^T\mathrm{x} = b\}$, therefore $a^T\mathrm{x}_1 = b$, $a^T\mathrm{x}_2 = b$ and $a^T\mathrm{x}_\mu = b$ for all $0 \leq \mu \leq 1$.

In consequence, we can enunciate the following corollary of convexity of a function, within a non-empty and non-singleton set defined by a set of linear equality constraints.

Corollary 4.1 (Convex function on a set of linear equality constraints) *Let $\mathcal{E}$ denote a non-empty and non-singleton set of $\mathbb{R}^d$ defined by l linear equality constraints such that $\mathcal{E} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A\mathrm{x} = \mathrm{b}\}$, where $A \in \mathbb{R}^{l \times d}$ is a matrix of constants and $\mathrm{b} \in \mathbb{R}^d$ is a real vector. If $f(\mathrm{x}) : \mathbb{R}^d \to \mathbb{R}$ is a convex function on a non-empty convex set Ω of $\mathbb{R}^d$, and $\mathcal{E} \cap \Omega$ is a non-empty and non-singleton convex subset of Ω, then $f(\mathrm{x})$ is a convex function on the subset $\mathcal{E} \cap \Omega$.*

Proof. See Appendix B. ■

We now add some definitions and lemmas for proving how the LCNM algorithm identifies a feasible region, which can contain at least a local minimum.

Definition 4.1 (Promising active sub-space) *Consider the problem $\mathcal{P}$ of minimizing an objective function $f(\mathrm{x})$ on $\mathbb{R}^d$ subject to $\mathrm{x} \in \mathcal{F}$, where $\mathcal{F}$ denotes a non-empty set of feasible points given by a set of linear inequality constraints such that $\mathcal{F} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A\mathrm{x} \geq \mathrm{b}\}$. We call a current promising active sub-space $\mathcal{F}_{pro}^{[q]}$ of the problem $\mathcal{P}$ at qth iteration of the LCNM algorithm, a feasible subset such that*

$$\mathcal{F}_{pro}^{[q]} \equiv \{\mathrm{x} \in \mathbb{R}^d \mid A_{act}\ S_{v_q}^{[q]} = \mathrm{b}_{act}\mathbf{1}_{v_q}^T \ \wedge \ A_{n-act}\ S_{v_q}^{[q]} > \mathrm{b}_{n-act}\mathbf{1}_{v_q}^T\}$$

where $A_{act}\ \mathrm{x} \geq \mathrm{b}_{act}$ is the subset of constraints of $\mathcal{F}$ that has been activated by the current simplex $S_{v_q}^{[q]}$ at the qth iteration of the LCNM algorithm, and $A_{n-act}\mathrm{x} > \mathrm{b}_{n-act}$ corresponds to the other subset of constraints that have not been activated by the current simplex.

We stress that such active sub-spaces are sub-spaces identified by the LCNM method during its search as being where the minimum might be. Thus the Definition 4.1 does not necessarily identify where the minimum really is, but a likely sub-space is one that is promising.

Conforming with the above definition, $\mathcal{F}_{pro}^{[q]}$ can be an empty subset at qth iteration of the LCNM algorithm, because it may be that no constraint has been activated by the current simplex. $\mathcal{F}_{pro}^{[q]}$ can also be an empty subset for all qth iterations of the LCNM algorithm, because no local minimum exists on the boundary of the feasible subset $\mathcal{F}$.

Definition 4.2 (Optimal active sub-space) *Consider the problem $\mathcal{P}$ of minimizing an objective function $f(\mathbf{x})$ on $\mathbb{R}^d$ subject to $\mathbf{x} \in \mathcal{F}$, where $\mathcal{F}$ denotes a non-empty set of feasible points given by a set of linear inequality constraints such that $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$, which at least contains a local minimum on the boundary of $\mathcal{F}$. We say that $\mathcal{F}^{[q]}_{pro}$ is optimal (minimum), which is denoted by $\mathcal{F}^{[q]}_{opt}$, if $\mathcal{F}^{[q]}_{opt}$ is a non-empty subset of $\mathcal{F}$ and*

$$\mathcal{F}^{[q]}_{opt} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A_{act}\ \mathbf{x}_{\min} = \mathbf{b}_{act} \ \wedge \ A_{n-act}\ \mathbf{x}_{\min} > \mathbf{b}_{n-act}\}$$

where $A_{act}\ \mathbf{x}_{\min} = \mathbf{b}_{act}$ represents the subset of linear constraints activated by the last qth simplex at sth stage of the LCNM algorithm, which has satisfied Inequality (3.39), and $A_{n-act}\mathbf{x}_{\min} > \mathbf{b}_{n-act}$ corresponds to the other subset of constraints that have not been activated by the current simplex.

It is worthwhile pointing out that if all the linear constraints of $\mathcal{F}$ have been activated by the current simplex, then there exists no constraint that satisfies $A_{n-act}\ \mathbf{x}_{\min} > \mathbf{b}^T_{n-act}$.

Let $v_q = d + 1 - r_q$ be the number of vertices at the qth iteration regarded as function of the collapsed degree r_q of the current simplex. Using this notation, we can write any current simplex at qth iteration as $S^{[q]}_{d+1-r_q}$ for explicitly indicating the collapsed degree r_q of the current simplex.

The problem of convergence highlighted by McKinnon (1998) are present when the minimum is an interior point. However when there is no local interior minimum then there should not be a problem. Though we do not give a definitive proof, the following argument indicates that the method should behave satisfactorily.

Theorem 4.3 (Moving into promising active sub-space) *Consider the problem $\mathcal{P}$ of minimizing a strictly convex and differentiable objective function $f(\mathbf{x})$ on $\mathbb{R}^d$ subject to $\mathbf{x} \in \mathcal{F}$, where $\mathcal{F}$ denotes a non-empty set of feasible points given by a set of k linear inequality constraints such that $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$. If the LCNM algorithm is applied to the problem $\mathcal{P}$ beginning with a simplex $S^{[q]}_{d+1-r_q}$ of r_q collapsed degree, where $0 < r_q < k$, and if there exists no local minimum at any internal point $\mathbf{x}_{int}$ of the current promising active sub-space of r_q active constraints, then the current simplex $S^{[q]}_{d+1-r_q}$ keeps moving in order to increase its collapsed degree.*

Proof. Due to the assumptions of the theorem, there exists a non-empty subset of active constraints at the qth iteration, wherein is the collapsed simplex $S^{[q]}_{d+1-r_q}$ and whose vertices belong to the intersection sub-space defined by the set of active constraints.

Moreover, due to the conditions of the theorem, no local minimum exists at any internal point of the intersection sub-space. Thus, the current simplex $S^{[q]}_{d+1-r_q}$ is in what we have defined as promising active sub-space $\mathcal{F}^{[q]}_{pro}$. Given these assumptions and using Theorem 4.1, we can be assured that the simplex keeps moving for increasing its collapsed degree. Namely, $r_{q+h} \geq r_q$ if h LCNM iterations are applied to the simplex $S^{[q]}_{d+1-r_q}$ under the assumptions given by the theorem. ■

Theorem 4.3 allows us to explain the moving into what we have defined as optimal active sub-space, when no local minimum exists at any internal point of the feasible region $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$.

In addition, (McKinnon 1998) presents a family of functions of two variables, where convergence occurs to a non-stationary point when it is applied the NM method. The work developed by McKinnon proves the necessary conditions for what the author calls a repeated focused inside contraction (RFIC), where recurrent inside contraction operations with the best vertex (focus) remaining fixes. In consequence, the NM method converges to a not necessarily stationary point (focus) by repeated inside operations, if the conditions for the RFIC remain during the process.

These conditions are:

$$f_1 \leq f_2 \leq f_3 \leq f_{refl} \tag{4.27a}$$

$$f_1 \leq f_{cont} \leq f_2 \leq f_3 \tag{4.27b}$$

If Inequalities (4.27a) and (4.27b) are sequentially satisfied at each qth iteration, the NM method repeatedly applies inside contraction operations, so the best vertex is kept at each iteration. This recurrent process could occur in the LCNM method when it is applied to a constrained optimization problem.

Analysing the LCNM method, a RFIC operations can occur in a d-dimensional optimization problem, if Inequalities (4.28a) and (4.28b) are satisfied at each qth iteration of the LCNM method. Thus,

$$f_1 \leq f_2 \leq \cdots \leq f_{v_q-2} \leq f_{v_q-1} \leq f_{v_q} \leq f_{refl} \tag{4.28a}$$

$$f_1 \leq f_{cont} \leq f_{v_q-1} \tag{4.28b}$$

In this case $\mathbf{x}_1$ is the focus of the RFIC operations, which could be a non-stationary point of the objective function.

Furthermore, Inequality (4.28b) does not establish the relative value of f_{cont} with respect to $f_2, f_3, \ldots f_{v_q-2}$ for a RFIC to take place.

Now we shall include an extension for the LCNM method of a lemma established by Lagarias et al. (1998), who proposed the following no occurring shrink operation, if the objective function is strictly convex.

Theorem 4.4 (Non-shrink operation) *Consider the problem $\mathcal{P}$ of minimizing a strictly convex objective function $f(\mathbf{x})$ on $\mathbb{R}^d$ subject to $\mathbf{x} \in \mathcal{F}$, where $\mathcal{F}$ denotes a non-empty set of feasible points given by a set of k linear inequality constraints such that $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$. If the LCNM algorithm is applied to the problem $\mathcal{P}$ with a simplex $S^{[q]}_{d+1-r_q}$ of r_q collapsed degree, where $0 \leq r_q \leq k$, then a non-shrink operation will be attempted by the LCNM algorithm.*

Proof. From Corollary 4.1, we can affirm that any $S^{[q]}_{d+1-r_q}$ will always be on a convex region at any qth iteration. Besides, the LCNM algorithm reaches its shrink step (step 9) only, when an inside contraction is not accepted at step 8. Due to Equation 4.4, an inside contraction operation is defined when $\theta = -\beta$, thus an inside contraction is given by $\mathbf{x}^q_{cont} = (1-\beta)\mathbf{x}^q_{cen} + \beta\mathbf{x}^q_{v_q}$. Since $f(\mathbf{x})$ is strictly convex on the feasible region $\mathcal{F}$ then we can be assured that $f\left[\mathbf{x}^q_{cont}\right] < (1-\beta)f\left(\mathbf{x}^q_{cen}\right) + \beta f\left(\mathbf{x}^q_{v_q}\right)$, for $0 < \beta < 1$, and because Lemma 4.3 establishes that if $f(\mathbf{x})$ is a strictly convex objective function on $\mathbb{R}^d$, then $f\left(\mathbf{x}^q_{cen}\right) < f\left(\mathbf{x}^q_{v_q}\right)$. Thus,

$$f\left(\mathbf{x}^q_{cont}\right) < (1-\beta)f\left(\mathbf{x}^q_{cen}\right) + \beta f\left(\mathbf{x}^q_{v_q}\right) < (1-\beta)f\left(\mathbf{x}^q_{v_q}\right) + \beta f\left(\mathbf{x}^q_{v_q}\right) = f\left(\mathbf{x}^q_{v_q}\right).$$

Hence, if the LCNM algorithm is applied to the problem $\mathcal{P}$ with a simplex $S^{[q]}_{d+1-r_q}$ of r_q collapsed degree and the LCNM algorithm reaches the step 8 at qth iteration, then $\mathbf{x}^q_{cont}$ is accepted and in consequence a shrink operation does not occur at any qth iteration. ∎

4.2.3 Case of two dimensions strictly convex function and a linear constraint

We herein shall consider the problem $\mathcal{P}$ of minimizing a strictly convex objective function $f(\mathbf{x})$ on $\mathbb{R}^2$ subject to $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^2 \mid \mathbf{a}^T\mathbf{x} \geq b\}$, where $\mathbf{a}^T = (a_1, a_2)$ is a vector of coefficients and b is a scalar. Moreover, $\nabla f(\mathbf{x}_{int}) \neq \mathbf{0}_d$ for all $\mathbf{x}_{int}$ in the interior of $\mathcal{F}$.

Through this case we shall study how an initial simplex is moved by the LCNM algorithm and how it is transformed when approaching the boundary of the feasible region, when no local minimum exists at any internal point of the feasible region. In this case study, we have proved that the vertex $\mathbf{x}_1^q$ of a qth simplex reaches the boundary of the feasible region within a finite number of iterations, if the conditions of the example are obviously satisfied during the optimization.

Suppose that we apply the LCNM algorithm to our problem with an initial point $\mathbf{x}_{initial}$ for building an ordered initial simplex $S_3^{[0]} = \left[\mathbf{x}_1^0 : \mathbf{x}_2^0 : \mathbf{x}_3^0\right]$. Moreover, assume that a constrained expansion operation is attempted at each qth iteration and, as result of ordering the vertices by the value of $f(\mathbf{x}_i)$ for $i = 1, 2$ and 3, the next $(q+1)$th iteration produces the simplex given by $\mathbf{x}_1^{q+1} = \mathbf{x}_{new}^q$, $\mathbf{x}_2^{q+1} = \mathbf{x}_1^q$ and $\mathbf{x}_3^{q+1} = \mathbf{x}_2^q$ for $0 < q < q_{\max}$, where $q_{\max}$ is the maximum number of required iterations for reaching once any constraint boundary by any vertex of the simplex $S_3^{[q_{\max}]}$.

Suppose that for the qth iteration we have the ordered simplex $S_3^{[q]} = [\mathbf{x}_1^q : \mathbf{x}_2^q : \mathbf{x}_3^q]$. Using Equation (4.9) we have

$$\mathbf{x}_1^{q+1} = \mathbf{x}_{new}^q = S_3^{[q]} \left[\frac{1+\theta}{2}, \frac{1+\theta}{2}, -\theta\right]^T$$

and given that $\mathbf{x}_2^{q+1} = \mathbf{x}_1^q$ and $\mathbf{x}_3^{q+1} = \mathbf{x}_2^q$ for $0 < q < q_{\max}$ due to Lemma 4.4, we can express

$$S_3^{[q+1]} = S_3^{[q]} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} + S_3^{[q]} \left[\frac{1+\theta}{2}, \frac{1+\theta}{2}, -\theta\right]^T [1, 0, 0] \tag{4.29}$$

Rearranging the terms of Equation (4.29) and using matrix notation, we have

$$S_3^{[q+1]} = S_3^{[q]} T(\theta) \tag{4.30}$$

where $T(\theta) = \begin{bmatrix} \frac{1+\theta}{2} & 1 & 0 \\ \frac{1+\theta}{2} & 0 & 1 \\ -\theta & 0 & 0 \end{bmatrix}$ is the called transition matrix.

From Equation (4.30), we obtain

$$S_3^{[q]} = S_3^{[0]} \left[T(\theta)\right]^q \quad \text{and} \quad S_3^{[q+n]} = S_3^{[q]} \left[T(\theta)\right]^n \tag{4.31}$$

Equation (4.31) permits us to compute any simplex $S_3^{[q]}$ for $0 < q < q_{\max}$ under the conditions given, and we call it, *transformation equation.*

Using Equation (4.30) for $\theta = 2$ and an initial simplex $S_3^{[0]}$, we obtain the following results for $q = 0, 1, \ldots 5$,

$$S_3^{[q]} = \begin{cases} \begin{bmatrix} m & m & m+\nu \\ m-\nu & m & m \end{bmatrix} & q=0 \\ \begin{bmatrix} m-2\nu & m & m \\ m-\frac{3}{2}\nu & m-\nu & m \end{bmatrix} & q=1 \\ \begin{bmatrix} m-3\nu & m-2\nu & m \\ m-\frac{15}{4}\nu & m-\frac{3}{2}\nu & m-\nu \end{bmatrix} & q=2 \\ \begin{bmatrix} m-\frac{15}{2}\nu & m-3\nu & m-2\nu \\ m-\frac{47}{8}\nu & m-\frac{15}{4}\nu & m-\frac{3}{2}\nu \end{bmatrix} & q=3 \\ \begin{bmatrix} m-\frac{47}{4}\nu & m-\frac{15}{2}\nu & m-3\nu \\ m-\frac{183}{16}\nu & m-\frac{47}{8}\nu & m-\frac{15}{4}\nu \end{bmatrix} & q=4 \\ \begin{bmatrix} m-\frac{183}{8}\nu & m-\frac{47}{4}\nu & m-\frac{15}{2}\nu \\ m-\frac{591}{32}\nu & m-\frac{183}{16}\nu & m-\frac{47}{8}\nu \end{bmatrix} & q=5 \end{cases}$$

However, another manner for calculating $S_3^{[q]}$ is based on Equation 4.31 and the definition of an eigenvector (Noble and Daniel 1988), so $[T(\theta)]^q$ can be computed by the following equation.

$$[T(\theta)]^q = PD^qP^{-1} \tag{4.32}$$

where the matrix $P = [\mathbf{v}_1 : \mathbf{v}_2 : \mathbf{v}_3]$ is the matrix of eigenvectors of $T(\theta)$, and the qth power matrix $D^q = diag\,[\mu_1^q, \mu_2^q, \mu_3^q]$ represents the diagonal matrix defined by the eigenvalues μ_i of $T(\theta)$ for $i = 1, 2$ and 3.

Without loss of generality, suppose that $\nu = m/10$ and the parameters associated with the linear constraint are positive, that is, $a_1 > 0$, $a_2 > 0$ and $b > 0$. Besides, assume that $m > \max\left[\frac{b}{a_1}, \frac{b}{a_2}\right]$. Because the vertex $\mathbf{x}_1^q$ corresponds to the minimum value of the objective function, its residual are calculated with respect to the boundary of the linear constraint, and its values are given by the following equation

$$\breve{r}\left(\mathbf{x}_1^q\right) = \begin{cases} \max[0, -b + ma_1 + \frac{9}{10}ma_2] = -b + ma_1 + \frac{9}{10}ma_2 & q = 0 \\ \max[0, -b + \frac{4}{5}ma_1 + \frac{17}{20}ma_2] = -b + \frac{4}{5}ma_1 + \frac{17}{20}ma_2 & q = 1 \\ \max[0, -b + \frac{7}{10}ma_1 + \frac{5}{8}ma_2] = -b + \frac{7}{10}ma_1 + \frac{5}{8}ma_2 & q = 2 \\ \max[0, -b + \frac{1}{4}ma_1 + \frac{33}{80}ma_2] = -b + \frac{1}{4}ma_1 + \frac{33}{80}ma_2 & q = 3 \\ \max[0, -b - \frac{7}{40}ma_1 - \frac{23}{160}ma_2] = 0 & q = 4. \end{cases}$$

Because of Theorem 4.1, either constrained reflection step or constrained expansion step occurs during the application of the LCNM method, until the simplex collapses onto the linear constraint boundary.

Note that the residual $\breve{r}\left(\mathbf{x}_1^q\right)$ would become negative for $q = 4$ and any value of $a_1 > 0$, $a_2 > 0$, $b > 0$ and $m > \max\left[\frac{b}{a_1}, \frac{b}{a_2}\right]$, if the LCNM method did not make use of the LCP. This property of the LCNM method reduces quickly the number of vertices during the process of minimizing, when the objective function is strictly convex and there exists no minimum inside the feasible region. Therefore, the number of function evaluations could be decreased.

Although, this particular case does not prove the general situation, we observe that the LCNM method reduces the number of vertices in a few iterations, when the local minimum is on the boundary of the feasible region.

4.3 Convergence properties of a triangular simplex

In this section we shall study the properties of convergence through particular cases when we apply the LCNM algorithm to a three-vertex simplex. These cases would occur when the LCNM algorithm reduces the number of vertices to three, as consequence of collapse of a simplex in a constrained optimization problem of three or more dimensions. Furthermore, two-dimension constrained or unconstrained optimization problems also bring these cases, if no collapse is produced during the application of the LCNM algorithm.

4.3.1 Cyclic contraction operations

Suppose that we apply the LCNM algorithm to either a constrained problem or unconstrained problem such that for the hth iteration we have obtained an ordered three vertices simplex (triangle) $S_3^{[h]} = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right]$, where $\mathbf{x}_i^h$ is a d-dimensional column vector for $i = 1, 2, 3$. Furthermore, suppose that there exists a unique local minimum inside the simplex $S_3^{[h]}$ at hth iteration, and also assume that there will be no reduction of the number of vertices. In this particular case, suppose that there will only be sequential and cyclic contraction operations, such that:

$$\begin{array}{lll} \text{a) } f(\mathbf{x}_1^{h+1+3n}) \leq f(\mathbf{x}_{cont}^{h+3n}) \leq f(\mathbf{x}_3^{h+1+3n}) & \text{at } (h+1+3n)\text{th iteration} \\ \text{b) } f(\mathbf{x}_{cont}^{h+1+3n}) \leq f(\mathbf{x}_{cont}^{h+3n}) \leq f(\mathbf{x}_1^{h+2+3n}) & \text{at } (h+2+3n)\text{th iteration} \\ \text{c) } f(\mathbf{x}_{cont}^{h+2+3n}) \leq f(\mathbf{x}_{cont}^{h+1+3n}) \leq f(\mathbf{x}_{cont}^{h+3n}) & \text{at } (h+3+3n)\text{th iteration} \end{array} \tag{4.33}$$

where $n = 0, 1, \ldots$ represents the number of the cycle. Cyclic contraction operations can occur because no shrink steps are operated if the objective function $f(\mathbf{x})$ is strictly convex (See Theorem 4.4 on page 82) and, each constrained reflection point and constrained expansion point are not accepted by the LCNM algorithm.

If the conditions given by Inequality (4.33) are satisfied for all LCNM iterations, a simplex $S_3^{[h+3n]} = \left[\mathbf{x}_1^{h+3n} : \mathbf{x}_2^{h+3n} : \mathbf{x}_3^{h+3n}\right]$ is cyclically transformed as follows

$$\begin{array}{lllll} S_3^{[h+3n]} = & \left[\mathbf{x}_1^{h+3n} : \mathbf{x}_2^{h+3n} : \mathbf{x}_3^{h+3n}\right] & & \\ S_3^{[h+1+3n]} = & \left[\mathbf{x}_1^{h+1+3n} : \mathbf{x}_2^{h+1+3n} : \mathbf{x}_3^{h+1+3n}\right] & = & \left[\mathbf{x}_1^{h+3n} : \mathbf{x}_{cont}^{h+3n} : \mathbf{x}_2^{h+3n}\right] \\ S_3^{[h+2+3n]} = & \left[\mathbf{x}_1^{h+2+3n} : \mathbf{x}_2^{h+2+3n} : \mathbf{x}_3^{h+2+3n}\right] & = & \left[\mathbf{x}_{cont}^{h+1+3n} : \mathbf{x}_{cont}^{h+3n} : \mathbf{x}_1^{h+3n}\right] \\ S_3^{[h+3+3n]} = & \left[\mathbf{x}_1^{h+3+3n} : \mathbf{x}_2^{h+3+3n} : \mathbf{x}_3^{h+3+3n}\right] & = & \left[\mathbf{x}_{cont}^{h+2+3n} : \mathbf{x}_{cont}^{h+1+3n} : \mathbf{x}_{cont}^{h+3n}\right] \end{array} \tag{4.34}$$

Using the conditions given by Inequality (4.33) and the so-called transition matrix, we can represent simplices at each iteration as follows

$$\begin{array}{l} S_3^{[h+1+3n]} = S_3^{[h+3n]} T_1(-\beta) \\ S_3^{[h+2+3n]} = S_3^{[h+1+3n]} T_2(-\beta) \\ S_3^{[h+3+3n]} = S_3^{[h+2+3n]} T_3(-\beta) \end{array} \tag{4.35}$$

On rearranging Equations (4.35) we obtained

$$S_3^{[h+3+3n]} = S_3^{[h+3n]} T_1(-\beta) T_2(-\beta) T_3(-\beta) \quad \forall\ n = 0, 1, \ldots \tag{4.36}$$

where

$$T_1(-\beta) = \begin{bmatrix} 1 & \frac{1-\beta}{2} & 0 \\ 0 & \frac{1-\beta}{2} & 1 \\ 0 & \beta & 0 \end{bmatrix}, \; T_2(-\beta) = \begin{bmatrix} \frac{1-\beta}{2} & 0 & 1 \\ \frac{1-\beta}{2} & 1 & 0 \\ \beta & 0 & 0 \end{bmatrix} \text{ and } T_3(-\beta) = \begin{bmatrix} \frac{1-\beta}{2} & 1 & 0 \\ \frac{1-\beta}{2} & 0 & 1 \\ \beta & 0 & 0 \end{bmatrix}.$$

If we define $T_c(-\beta) = T_1(-\beta)T_2(-\beta)T_3(-\beta)$ then Equation (4.36) can be represented as

$$S_3^{[h+3+3n]} = S_3^{[h+3n]} T_c(-\beta) \;\; \forall \, n = 0, 1, \ldots \tag{4.37}$$

where

$$T_c(-\beta) = \begin{bmatrix} \frac{1}{8}\beta + \frac{5}{8}\beta^2 - \frac{1}{8}\beta^3 + \left(-\frac{1}{2}\beta + \frac{1}{2}\right)^2 + \frac{3}{8} & -\frac{1}{2}\beta + \left(-\frac{1}{2}\beta + \frac{1}{2}\right)^2 + \frac{1}{2} & -\frac{1}{2}\beta + \frac{1}{2} \\ \frac{1}{8}\beta - \frac{1}{8}\beta^2 - \frac{1}{8}\beta^3 + \left(-\frac{1}{2}\beta + \frac{1}{2}\right)^2 + \frac{1}{8} & \beta + \left(-\frac{1}{2}\beta + \frac{1}{2}\right)^2 & -\frac{1}{2}\beta + \frac{1}{2} \\ \frac{3}{4}\beta - \beta^2 + \frac{1}{4}\beta^3 & \frac{1}{2}\beta - \frac{1}{2}\beta^2 & \beta \end{bmatrix}.$$

From Equation (4.37) and using induction, it can be proved

$$S_3^{[h+3+3n]} = S_3^{[h]} T_c^{n+1}(-\beta) \tag{4.38}$$

It is worthwhile mentioning that the determinants of $T_c(-\beta)$ and $T_c^{n+1}(-\beta)$ are

$$det \; T_c(-\beta) = \beta^3, \quad det \; T_c^{n+1}(-\beta) = \beta^{3(n+1)}$$

Moreover, since $0 < \beta < 1$, then

$$\lim_{n \to \infty} \left[det \; T_c^{n+1}(-\beta)\right] = \lim_{n \to \infty} \left[\beta^{3(n+1)}\right] = 0, \quad \forall \, 0 < \beta < 1 \, . \tag{4.39}$$

Equation (4.39) represents a necessary condition of convergence of the LCNM algorithm when a sequential and cyclic contraction operation of three steps occurs on a three vertex simplex.

Note that the cycle of contraction operations is equal to three, that is, each three iterations is repeated a sequence of contraction operation that satisfies the conditions given by Equation (4.35).

In order to compute the $\lim_{n \to \infty} S_3^{[k+3+3n]}$, $T_c(-\beta)$ will be decomposed into its eigensystem PDP^{-1}, where $P = [\mathbf{v}_1 : \mathbf{v}_2 : \mathbf{v}_3]$ is the matrix given by the eigenvectors of $T_c(-\beta)$ and $D = diag\,[\mu_1, \mu_2, \mu_3]$ is a diagonal matrix formed by the respective eigenvalues of $T_c(-\beta)$. Therefore, $T_c^{n+1}(-\beta)$ can be expressed as

$$T_c^{n+1}(-\beta) = PD^{n+1}P^{-1} \tag{4.40}$$

Without losing of generality, we shall compute Equation (4.40) for $\beta = 1/2$. For this particular case, the matrices P, P^{-1} and D are given by

$$P = \begin{bmatrix} 2 & -\frac{3}{4} - \frac{1}{4}i\sqrt{23} & -\frac{3}{4} + \frac{1}{4}i\sqrt{23} \\ \frac{3}{2} & -\frac{1}{4} + \frac{1}{4}i\sqrt{23} & -\frac{1}{4} - \frac{1}{4}i\sqrt{23} \\ 1 & 1 & 1 \end{bmatrix},$$

$$P^{-1} = \begin{bmatrix} \frac{2}{9} & \frac{2}{9} & \frac{2}{9} \\ -\frac{1}{9} + \frac{7}{207}i\sqrt{23} & -\frac{1}{9} - \frac{11}{207}i\sqrt{23} & \frac{7}{18} + \frac{5}{414}i\sqrt{23} \\ -\frac{1}{9} - \frac{7}{207}i\sqrt{23} & -\frac{1}{9} + \frac{11}{207}i\sqrt{23} & +\frac{7}{18} - \frac{5}{414}i\sqrt{23} \end{bmatrix} \quad \text{and}$$

$$D^{n+1} = diag\left[\mu_1^{n+1}, \mu_2^{n+1}, \mu_3^{n+1}\right] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \left(\frac{45}{128} - \frac{1}{128}i\sqrt{23}\right)^{n+1} & 0 \\ 0 & 0 & \left(\frac{45}{128} + \frac{1}{128}i\sqrt{23}\right)^{n+1} \end{bmatrix}.$$

Since $\left|\frac{45}{128} \pm \frac{1}{128}i\sqrt{23}\right| = \frac{1}{4}\sqrt{2} < 1$, then $\lim_{n\to\infty}\left(\frac{45}{128} \pm \frac{1}{128}i\sqrt{23}\right)^{n+1} = 0$.
Thus, $\lim_{n\to\infty} D^{n+1}$ is given by

$$\lim_{n\to\infty} D^{n+1} = D^{\infty} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \text{and}$$

$$\lim_{n\to\infty} PD^{n+1}P^{-1} = PD^{\infty}P^{-1} = \begin{bmatrix} \frac{4}{9} & \frac{4}{9} & \frac{4}{9} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{2}{9} & \frac{2}{9} & \frac{2}{9} \end{bmatrix} \tag{4.41}$$

Using Equations (4.38) and (4.41), we obtain

$$\lim_{n\to\infty} S_3^{[h]}T_c^{n+1}(-1/2) = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right] \begin{bmatrix} \frac{4}{9} & \frac{4}{9} & \frac{4}{9} \\ \frac{3}{9} & \frac{3}{9} & \frac{3}{9} \\ \frac{2}{9} & \frac{2}{9} & \frac{2}{9} \end{bmatrix},$$

which proves that when sequential and cyclic contraction operations are repeatedly performed under the assumptions and conditions given by Inequality (4.33), the three vertices of the simplex $S_3^{[h]} = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right]$ converge to the point

$\mathbf{x}_{\min} = \frac{4}{9}\mathbf{x}_1^h + \frac{3}{9}\mathbf{x}_2^h + \frac{2}{9}\mathbf{x}_3^h$.

According to the conditions given by Inequality (4.33), within each nth cycle: the vertex $\mathbf{x}_1^{h+3n}$ remains during three iterations, $\mathbf{x}_2^{h+3n}$ remains at two iterations and $\mathbf{x}_3^{h+3n}$ is at one iteration only. This can explain the reason why the minimum point is a weighted average of the number of iterations that permanentizes each vertex within each cycle.

Note that the determinant of $\lim_{n\to\infty} T_c^{n+1}(-1/2) = 0$ verifies the necessary condition given by Equation (4.39).

Since each vertex of the simplex $S_3^{[h+3+3n]}$ approaches the minimum point $\mathbf{x}_{\min}$ as the result of successive transformations yielded by Equation (4.37), the behaviour of one of these vertices of $S_3^{[h+3+3n]}$ can be studied through Equation (4.38).

Using Equation (4.40) we compute $T_c^{n+1}(-1/2)$, so

$$T_c^{n+1}(-1/2) = [\mathbf{t}_1 : \mathbf{t}_2 : \mathbf{t}_3] = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}, \tag{4.42}$$

where

$$\mathbf{t}_1 = \begin{bmatrix} -\mu_2^{n+1}\left(-\frac{5}{18} - i\frac{\sqrt{23}}{414}\right) - \mu_3^{n+1}\left(-\frac{5}{18} + i\frac{\sqrt{23}}{414}\right) + \frac{4}{9} \\ -\mu_2^{n+1}\left(\frac{1}{6} + i\frac{5\sqrt{23}}{138}\right) - \mu_3^{n+1}\left(\frac{1}{6} - i\frac{5\sqrt{23}}{138}\right) + \frac{1}{3} \\ \mu_2^{n+1}\left(-\frac{1}{9} + i\frac{7\sqrt{23}}{207}\right) + \mu_3^{n+1}\left(-\frac{1}{9} - i\frac{7\sqrt{23}}{207}\right) + \frac{2}{9} \end{bmatrix},$$

$$\mathbf{t}_2 = \begin{bmatrix} -\mu_2^{n+1}\left(\frac{2}{9} - i\frac{14\sqrt{23}}{207}\right) - \mu_3^{n+1}\left(\frac{2}{9} + i\frac{14\sqrt{23}}{207}\right) + \frac{4}{9} \\ -\mu_2^{n+1}\left(-\frac{1}{3} + i\frac{\sqrt{23}}{69}\right) - \mu_3^{n+1}\left(-\frac{1}{3} - i\frac{\sqrt{23}}{69}\right) + \frac{1}{3} \\ \mu_2^{n+1}\left(-\frac{1}{9} - i\frac{11\sqrt{23}}{207}\right) + \mu_3^{n+1}\left(-\frac{1}{9} + i\frac{11\sqrt{23}}{207}\right) + \frac{2}{9} \end{bmatrix},$$

$$\mathbf{t}_3 = \begin{bmatrix} -\mu_2^{n+1}\left(\frac{2}{9} + i\frac{22\sqrt{23}}{207}\right) - \mu_3^{n+1}\left(\frac{2}{9} - i\frac{22\sqrt{23}}{207}\right) + \frac{4}{9} \\ -\mu_2^{n+1}\left(\frac{1}{6} - i\frac{13\sqrt{23}}{138}\right) - \mu_3^{n+1}\left(\frac{1}{6} + i\frac{13\sqrt{23}}{138}\right) + \frac{1}{3} \\ \mu_2^{n+1}\left(\frac{7}{18} + i\frac{5\sqrt{23}}{414}\right) + \mu_3^{n+1}\left(\frac{7}{18} - i\frac{5\sqrt{23}}{414}\right) + \frac{2}{9} \end{bmatrix},$$

with $\mu_2 = \frac{45}{128} - \frac{1}{128}i\sqrt{23}$ and $\mu_3 = \frac{45}{128} + \frac{1}{128}i\sqrt{23}$.

Moreover, we have a triangular simplex in a d-dimensional Euclidean space represented by $S_3^{[h]} = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right]$.

From Equation (4.38) we obtain

$$S_3^{[h+3+3n]} = S_3^{[h]} T_c^{n+1}(-\beta) = \begin{bmatrix} t_{11}\mathbf{x}_1^h + t_{21}\mathbf{x}_2^h + t_{31}\mathbf{x}_3^h \\ t_{12}\mathbf{x}_1^h + t_{22}\mathbf{x}_2^h + t_{32}\mathbf{x}_3^h \\ t_{13}\mathbf{x}_1^h + t_{23}\mathbf{x}_2^h + t_{33}\mathbf{x}_3^h \end{bmatrix}^T \quad (4.43)$$

To study the rate of convergence, we shall examine the behaviour the vertices of $S_3^{[h+3+3n]}$ through the vectors $\mathbf{t}_1$, $\mathbf{t}_2$ and $\mathbf{t}_3$.

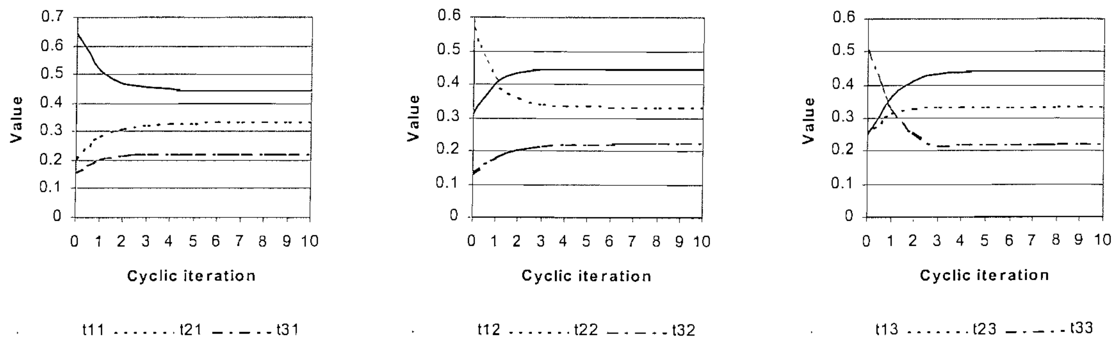


Figure 4.1: Graph of $\mathbf{t}_1$, $\mathbf{t}_2$ and $\mathbf{t}_3$ of $T_c^{n+1}(-1/2)$

Because $|\mu_2| < 1$ and $|\mu_3| < 1$, the simplex evidently converges to $\mathbf{x}_{\min} = \frac{4}{9}\mathbf{x}_1^h + \frac{3}{9}\mathbf{x}_2^h + \frac{2}{9}\mathbf{x}_3^h$.

Figure 4.1 displays the behaviour of the components of the vectors $\mathbf{t}_i(n)$ for each n cyclic iteration, where each component shows a behaviour of order $O(\xi^{-n})$ with $0 < \xi < 1$. For this particular case, we have that $\mathbf{t}_1(n)$, $\mathbf{t}_2(n)$ and $\mathbf{t}_3(n)$ simultaneously approach to the vector $\left(\frac{4}{9}, \frac{3}{9}, \frac{2}{9}\right)^T \approx (0.444\,44, 0.333\,33, 0.222\,22)^T$ at $n = 6$ cyclic iterations, that is, after six cyclic contraction operations of three iterations each, the LCNM algorithm moves the vertex $\mathbf{x}_1^{h+3+3n}$ of the simplex towards $\mathbf{x}_{\min} = \frac{4}{9}\mathbf{x}_1^h + \frac{3}{9}\mathbf{x}_2^h + \frac{2}{9}\mathbf{x}_3^h$.

This fact assures a convergence of order $O(\xi^{-(n+1)})$, if the conditions given by Inequality (4.33) have been held for the iterations $h < q < q_{stop}$, where q_{stop} is the last iteration carried out.

4.3.2 Repeated focused inside contraction operation at a minimum

The definition of RFIC was introduced by McKinnon (1998) for proving that there exists a kind of strictly convex function family defined over $\mathbb{R}^2$ that converges to a non-stationary point the NM method is applied. However, this approach can be used for estimating the maximal rate of convergence when we suppose that one of the vertices of a triangular simplex is located on the minimum of the objective function.

In this case, assume that we apply the LCNM algorithm to either a constrained problem or an unconstrained optimization problem such that after hth iteration, one of the vertices of the simplex $S_3^{[h]} = [\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h]$ is located at the minimum point of a strictly convex objective function $f(\mathbf{x})$, this is, $\mathbf{x}_{\min} = \mathbf{x}_1^h$, where $\mathbf{x}_i^h$ is a d-dimensional column vector for $i = 1, 2, 3$.

Without further assumptions, we can be assured that it only takes place an *inside contraction operation* at each $h < q \leq q_{stop}$. Because $\mathbf{x}_1^h$ is located at the minimum and the objective function is strictly convex, it can also be affirmed that there are inside contraction operations. Moreover, the shrink operation does not take place during the process of convergence (See Theorem 4.4 on page 82). Thus the transformations that are held by each qth simplex $S_3^{[q]}$ can be represented by Equation (4.44),

$$S_3^{[h+n]} = S_3^{[h]} T_1^n(-\beta) \quad \forall\, 0 \leq n \leq q_{stop} - h\ , \tag{4.44}$$

where $T_1(-\beta) = \begin{bmatrix} 1 & \frac{1-\beta}{2} & 0 \\ 0 & \frac{1-\beta}{2} & 1 \\ 0 & \beta & 0 \end{bmatrix}$ and q_{stop} represents the iteration wherein is satisfied the stopping rule.

Without losing of generality, we study the case $\beta = 1/2$, thus $T_1(-1/2) = \begin{bmatrix} 1 & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & 1 \\ 0 & \frac{1}{2} & 0 \end{bmatrix}$.

Using the eigensystem of $T_1(-1/2)$, we obtain $T_1^n(-1/2) = PD^nP^{-1}$

$$PD^nP^{-1} = \begin{bmatrix} 1 & \frac{\sqrt{33}-5}{4} & -\frac{\sqrt{33}+5}{4} \\ 0 & \frac{1-\sqrt{33}}{4} & \frac{1+\sqrt{33}}{4} \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \left(\frac{1-\sqrt{33}}{8}\right)^n & 0 \\ 0 & 0 & \left(\frac{1+\sqrt{33}}{8}\right)^n \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 0 & \frac{-2\sqrt{33}}{33} & \frac{\sqrt{33}+33}{66} \\ 0 & \frac{2\sqrt{33}}{33} & \frac{33-\sqrt{33}}{66} \end{bmatrix}, \tag{4.45}$$

which will be denoted by

$$T_1^n(-1/2) = [\mathbf{t}_1 : \mathbf{t}_2 : \mathbf{t}_3] = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}.$$

Using this notation and Equation (4.45), each $(h+n)$th simplex can be represented as

$$S_3^{[h+n]} = S_3^{[h]} T_1^n(-\beta) = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right] [\mathbf{t}_1 : \mathbf{t}_2 : \mathbf{t}_3] = \begin{bmatrix} t_{11}\mathbf{x}_1^h + t_{21}\mathbf{x}_2^h + t_{31}\mathbf{x}_3^h \\ t_{12}\mathbf{x}_1^h + t_{22}\mathbf{x}_2^h + t_{32}\mathbf{x}_3^h \\ t_{13}\mathbf{x}_1^h + t_{23}\mathbf{x}_2^h + t_{33}\mathbf{x}_3^h \end{bmatrix}^T, \quad (4.46)$$

where

$$\mathbf{t}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{t}_2 = \begin{bmatrix} \sqrt{33}\left(\frac{5-\sqrt{33}}{66}\right)\left(\frac{1-\sqrt{33}}{8}\right)^n - \sqrt{33}\left(\frac{5+\sqrt{33}}{66}\right)\left(\frac{1+\sqrt{33}}{8}\right)^n + 1 \\ \sqrt{33}\left(\frac{\sqrt{33}-1}{66}\right)\left(\frac{1-\sqrt{33}}{8}\right)^n + \sqrt{33}\left(\frac{1+\sqrt{33}}{66}\right)\left(\frac{1+\sqrt{33}}{8}\right)^n \\ \frac{2}{33}\sqrt{33}\left(\frac{1+\sqrt{33}}{8}\right)^n - \frac{2}{33}\sqrt{33}\left(\frac{1-\sqrt{33}}{8}\right)^n \end{bmatrix} \quad \text{and}$$

$$\mathbf{t}_3 = \begin{bmatrix} \left(\frac{\sqrt{33}-5}{4}\right)\left(\frac{33+\sqrt{33}}{66}\right)\left(\frac{1-\sqrt{33}}{8}\right)^n - \left(\frac{\sqrt{33}+5}{4}\right)\left(\frac{33-\sqrt{33}}{66}\right)\left(\frac{1+\sqrt{33}}{8}\right)^n + 1 \\ \left(\frac{1-\sqrt{33}}{4}\right)\left(\frac{33+\sqrt{33}}{66}\right)\left(\frac{1-\sqrt{33}}{8}\right)^n + \left(\frac{1+\sqrt{33}}{4}\right)\left(\frac{33-\sqrt{33}}{66}\right)\left(\frac{1+\sqrt{33}}{8}\right)^n \\ \left(\frac{33-\sqrt{33}}{66}\right)\left(\frac{1+\sqrt{33}}{8}\right)^n + \left(\frac{33+\sqrt{33}}{66}\right)\left(\frac{1-\sqrt{33}}{8}\right)^n \end{bmatrix}.$$

Observe that $\lim_{n\to\infty} T_1^n(-1/2) = (1,0,0)^T \mathbf{1}_3^T$ guarantees the convergence to the point $\mathbf{x}_1^h$.

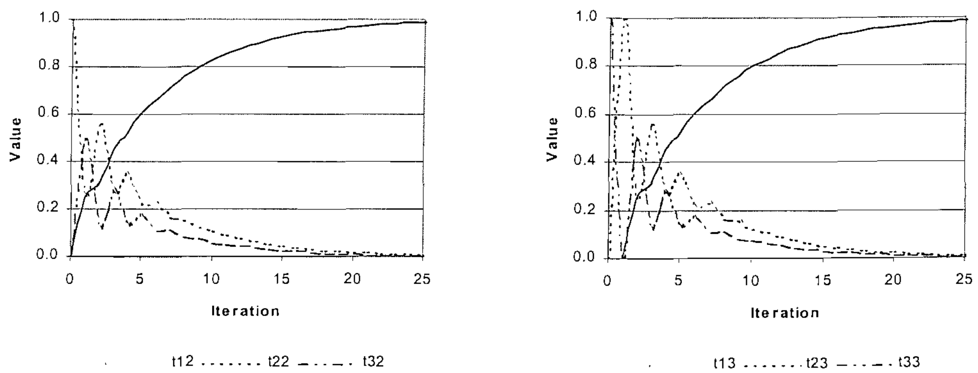


Figure 4.2: Graph of $\mathbf{t}_2$ and $\mathbf{t}_3$ of $T_1^n(-1/2)$.

Figure 4.2 depicts the behaviour of each component of $\mathbf{t}_2$ and $\mathbf{t}_3$ during the application of the LCNM method. Note that t_{12} and t_{13} converge to one exponentially, whilst the rest of the components of $\mathbf{t}_2$ and $\mathbf{t}_3$ converge to zero in an exponentially oscillatory way. This fact allows one to explain the convergence of the LCNM method in this particular situation.

4.3.3 Alternative operations at a minimum

In this case, we shall study the performance of the LCNM method, when sequential and cyclic alternative operations of contraction and reflection are carried out over a triangular simplex $S_3^{[h]} = \left[\mathbf{x}_1^h : \mathbf{x}_2^h : \mathbf{x}_3^h\right]$ in a d-dimensional Euclidean space, where $\mathbf{x}_i^h$ is a d-dimensional column vector for $i = 1, 2, 3$. Suppose that the vertex $\mathbf{x}_1^h$ is only located on the minimum point $\mathbf{0}_d$ after h iterations, and the distances $\left\|\mathbf{x}_1^h - \mathbf{x}_2^h\right\| > 0$ and $\left\|\mathbf{x}_1^h - \mathbf{x}_3^h\right\| > 0$ at the hth iteration. In this particular case, the objective function is the quadratic function $f(\mathbf{x}_d) = \|\mathbf{x}_d\|$. We also assume that there will be no reduction of the number of vertices.

Since the objective function is strictly convex, non-shrink operations will occur during the performance of the method (See Theorem 4.4 on page 82). If these sequential and cyclic alternative operations are performed during the process, we can affirm that

$$S_3^{[h+2n]} = S_3^{[h]} T_c^n(-\beta, \alpha), \text{ where} \tag{4.47}$$

$$T_c(-\beta,\alpha) = T_1(-\beta)T_2(\alpha) = \begin{bmatrix} 1 & \frac{1-\beta}{2} & 0 \\ 0 & \frac{1-\beta}{2} & 1 \\ 0 & \beta & 0 \end{bmatrix} \begin{bmatrix} 1 & \frac{1+\alpha}{2} & 0 \\ 0 & \frac{1+\alpha}{2} & 1 \\ 0 & -\alpha & 0 \end{bmatrix} = \begin{bmatrix} 1 & \frac{3\alpha-\beta-\alpha\beta+3}{4} & \frac{1-\beta}{2} \\ 0 & \frac{1-3\alpha-\beta-\alpha\beta}{4} & \frac{1-\beta}{2} \\ 0 & \frac{\beta+\alpha\beta}{2} & \beta \end{bmatrix},$$

which represents a contraction operation and a reflection operation within each mth cyclic of two iterations.

If $\beta = 1/2$ and $\alpha = 1$, we obtain

$$T_c(-1/2, 1) = \begin{bmatrix} 1 & \frac{5}{4} & \frac{1}{4} \\ 0 & -\frac{3}{4} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Using the eigensystem of $T_c(-1/2, 1)$, we obtain $T_c^n(-1/2, 1) = PD^nP^{-1}$ expressed by,

$$P = \begin{bmatrix} 1 & \frac{1}{4}\sqrt{33}+\frac{1}{4} & -\frac{1}{4}\sqrt{33}+\frac{1}{4} \\ 0 & -\frac{1}{4}\sqrt{33}-\frac{5}{4} & \frac{1}{4}\sqrt{33}-\frac{5}{4} \\ 0 & 1 & 1 \end{bmatrix},\ P^{-1} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & -\frac{2}{33}\sqrt{33} & -\frac{5}{66}\sqrt{33}+\frac{1}{2} \\ 0 & \frac{2}{33}\sqrt{33} & \frac{5}{66}\sqrt{33}+\frac{1}{2} \end{bmatrix} \text{ and}$$

$$D^n = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \left(-\frac{1}{8}\sqrt{33}-\frac{1}{8}\right)^n & 0 \\ 0 & 0 & \left(\frac{1}{8}\sqrt{33}-\frac{1}{8}\right)^n \end{bmatrix}.$$

We define $T_c^m(-1/2,1) = [\mathbf{t}_1 : \mathbf{t}_2 : \mathbf{t}_3] = \begin{bmatrix} t_{11} & t_{12} & t_{13} \\ t_{21} & t_{22} & t_{23} \\ t_{31} & t_{32} & t_{33} \end{bmatrix}$, where

$$\mathbf{t}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},\ \mathbf{t}_2 = \begin{bmatrix} -\frac{33+\sqrt{33}}{66}\left(-\frac{\sqrt{33}+1}{8}\right)^n + \frac{\sqrt{33}-33}{66}\left(\frac{\sqrt{33}-1}{8}\right)^n + 1 \\ \left(\frac{33-5\sqrt{33}}{66}\right)\left(\frac{\sqrt{33}-1}{8}\right)^n + \left(\frac{33+5\sqrt{33}}{66}\right)\left(-\frac{\sqrt{33}+1}{8}\right)^n \\ -\frac{2\sqrt{33}}{33}\left(-\frac{\sqrt{33}+1}{8}\right)^n + \frac{2\sqrt{33}}{33}\left(\frac{\sqrt{33}-1}{8}\right)^n \end{bmatrix} \text{ and}$$

$$\mathbf{t}_3 = \begin{bmatrix} \left(-\frac{7\sqrt{33}+33}{66}\right)\left(\frac{\sqrt{33}-1}{8}\right)^n + \left(\frac{7\sqrt{33}-33}{66}\right)\left(-\frac{1+\sqrt{33}}{8}\right)^n + 1 \\ \left(-\frac{\sqrt{33}}{33}\right)\left(-\frac{1+\sqrt{33}}{8}\right)^n + \left(\frac{\sqrt{33}}{33}\right)\left(\frac{\sqrt{33}-1}{8}\right)^n \\ \left(\frac{33-5\sqrt{33}}{66}\right)\left(-\frac{1+\sqrt{33}}{8}\right)^n + \left(\frac{33+5\sqrt{33}}{66}\right)\left(\frac{\sqrt{33}-1}{8}\right)^n \end{bmatrix}.$$

Observe that $\lim_{n\to\infty} T_c^n(-1/2,1) = (1,0,0)^T\,\mathbf{1}_3^T$ assures the convergence to the point $\mathbf{x}_1^h$.

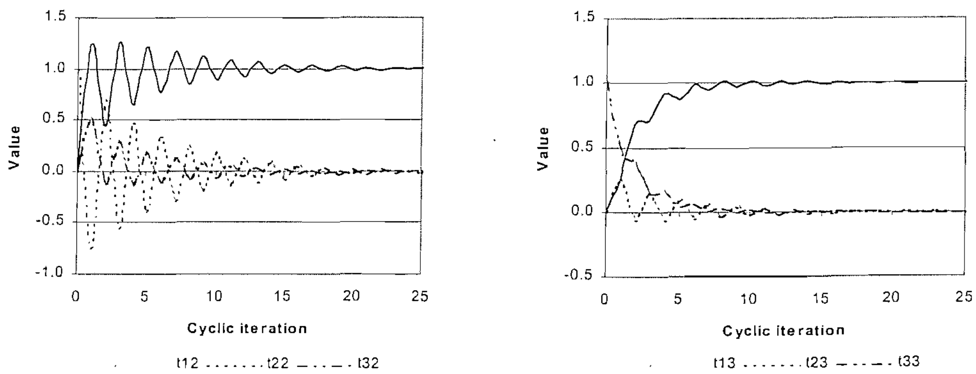


Figure 4.3: Graph of $\mathbf{t}_2$ and $\mathbf{t}_3$ of $T_c^n(-1/2,1)$

Figure 4.3 shows the shape how the LCNM algorithm converges when sequential and cyclic alternative operations of contraction and reflection occur over a triangular simplex.

The figure illustrates the exponential and oscillatory convergence of each component of $\mathbf{t}_2$, whilst the components of $\mathbf{t}_3$ converge exponentially and with small oscillations.

4.4 Case of constrained linear objective function

Here we shall study how the LCNM method can be dramatically slow when it is applied to a minimization Problem $\mathcal{P}$ of a linear objective function subject to two linear constraints symmetrically located with respect to the line $x_2 = x_1$. A mathematical formulation of this problem is as follows,

$$\min_{\mathbf{x}\in\mathbb{R}^2} f(\mathbf{x}) = \min_{\mathbf{x}\in\mathbb{R}^2} \bar{\mathbf{c}}^T\mathbf{x} = \min_{\mathbf{x}\in\mathbb{R}^2} (\bar{c}_1 x_1 + \bar{c}_2 x_2) \tag{4.48a}$$

subject to

$$\begin{aligned} l_1: &\quad \bar{b}x_1 - \bar{a}x_2 \geq 0 \\ l_2: &\quad -\bar{a}x_1 + \bar{b}x_2 \geq 0 \end{aligned} \tag{4.48b}$$

where $\bar{c}_1 > 0$, $\bar{c}_2 > 0$, and $\bar{b} > \bar{a} > 0$.

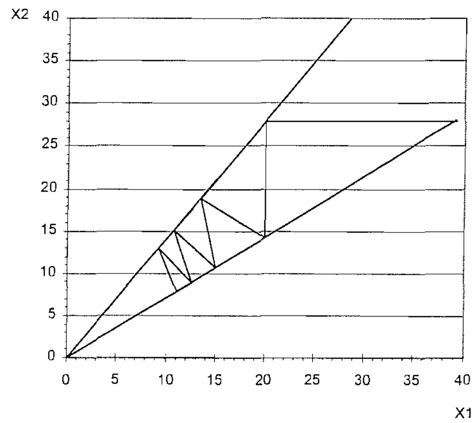


Figure 4.4: Case whereby the simplex collapses.

Figure 4.4 depicts the case when the LCNM method is applied to a linear objective function with parameters $\bar{c}_1 = 1.1$ and $\bar{c}_2 = 1$, subject to constraints given by Equation (4.48b) with parameters $\bar{b} = 1.4$ and $\bar{a} = 1$. For this numerical example we considered the point $(20, 28)^T$ as the initial point for building the entire simplex S_3^0. Note that the simplex S_3^8 collapses onto the boundary of the constraint l_2.

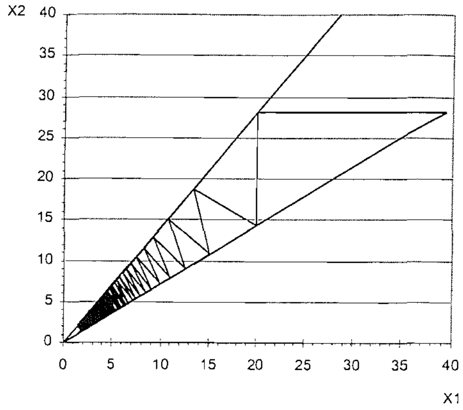


Figure 4.5: Case whereby the simplex does not collapse.

By contrast, Figure 4.5 displays 300 iterations of the LCNM algorithm. As can be seen from the graph, no simplex S_3^q collapses onto any boundary during the first 300 iterations, when the coefficient $\bar{c}_1$ was fixed to 1.850 7, and the rest of the parameters were kept to the same value of the previous case. From the numerical report of this case, we obtained at 300th iteration the following simplex

$$S_3^{300} = \left[\mathbf{x}_1^{300} : \mathbf{x}_2^{300} : \mathbf{x}_3^{300}\right] = \begin{bmatrix} 1.849588979 & 1.461849766 & 1.855732942 \\ 1.321134985 & 2.046589673 & 1.32552353 \end{bmatrix},$$

whose maximum edge length does not satisfy the stopping rule at the first stage of the LCNM algorithm and any vertex has not reached the local minimum.

The example illustrates the possibility of conditions where the triangular simplex does not collapse at any iteration. The problem is analysed in detail in the next subsection. Thus it is possible in principle for the LCNM algorithm to become expensive (Cheng 2003b).

It should be stressed however, that though this is a theoretical possibility, it seems unlikely that it would be a serious practical problem as the conditions have to be specific for the problem to occur.

4.4.1 A canonical form for Problem $\mathcal{P}$

Here we shall study a *canonical* form shown by Cheng (2003a) for a two-dimensional Linear Programming Problem (LPP).

Problem 4.1 (Linear Programming Problem) *Consider the following two-variable LPP*

$$\min_{\mathbf{x}\in\mathbb{R}^2} f(\mathbf{x}) = \min_{\mathbf{x}\in\mathbb{R}^2} c(x_1 + x_2) \tag{4.49}$$

subject to

$$\begin{aligned} l_1 &: \quad a_2x_1 - a_1x_2 \geq 0, \\ l_2 &: -a_1x_1 + a_2x_2 \geq 0, \end{aligned} \tag{4.50}$$

where

$$a_2 > a_1 > 0 \quad \textit{and} \quad c > 0. \tag{4.51}$$

The feasible region given by Inequalities (4.50) is a two-dimensional cone in the positive orthant if $a_2 > a_1 > 0$. Note that if the coefficient of the objective function is positive, the constrained local minimum will occur at the vertex of the cone, which is located at the origin. Though we only consider two constraints, the problem is a completely general (linear) two-dimensional one, in the sense that the problem reflects the geometry near the optimal vertex whatever the nature of the original problem.

Figure 4.6 illustrates a constrained reflection operation that takes place at an iteration of the LCNM method.

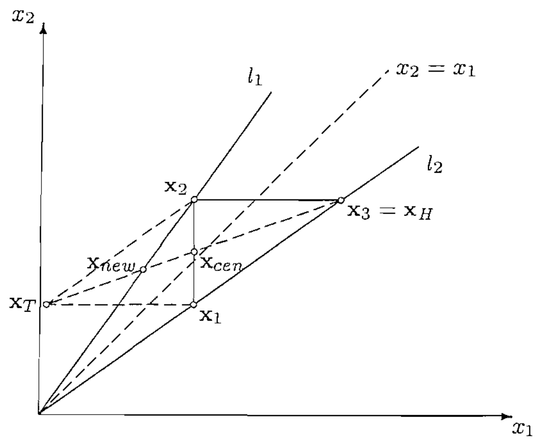


Figure 4.6: Constrained reflection operation

Figure 4.7 depicts the feasible cone-region for Problem 4.1. For clarity we have assumed a *canonical* form for Problem 4.1 in the sense that the two linear constraints are symmetrically located on both side of the line $x_2 = x_1$, with the coefficients of x_1 and x_2 in the objective being equal, that is

$$f(\mathbf{x}) = c(x_1 + x_2) \quad \forall \mathbf{x} \in \mathbb{R}^2 \text{ and } c > 0.$$

The two linear constraints, $\mathbf{l}_1^T\mathbf{x} \geq 0$ and $\mathbf{l}_2^T\mathbf{x} \geq 0$, are defined by direction vectors $\mathbf{l}_1^T = (a_2, -a_1)$ and $\mathbf{l}_2^T = (-a_1, a_2)$.

The problem however is equivalent to a completely general (linear) two-dimensional one where just two active constraints will determine the optimal vertex whatever the number of constraints in the original problem.

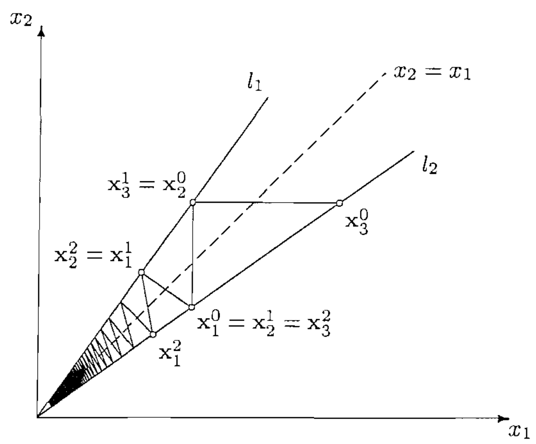


Figure 4.7: Sequence of the simplices S^q

To see that there is no loss of generality, suppose Problem 4.1 with variables $\mathbf{y}^T = (y_1, y_2)$, we therefore have the following problem

Problem 4.2 (Generalized Linear Programming Problem)

$$\min_{\mathbf{y}\in\mathbb{R}^2} f(\mathbf{y}) = \min_{\mathbf{y}\in\mathbb{R}^2} \mathbf{c}^T\mathbf{y}, \tag{4.52}$$

subject to

$$\mathbf{e}_1^T\mathbf{y} \geq 0 \textit{ and } \mathbf{e}_2^T\mathbf{y} \geq 0, \tag{4.53}$$

where $\mathbf{c}^T = (c_1, c_2)$ *and* $\mathbf{e}_j^T = (e_{1j}, e_{2j})$ *for* $j = 1, 2$.

We allow the direction vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ to be not necessarily symmetrically placed on either side of the line $x_2 = x_1$. However, we shall adopt the convention that the feasible region lies in the angle θ, with $0 < \theta < \pi$, between them, measured in the rotational sense of going from $\mathbf{e}_1$ to $\mathbf{e}_2$. This is equivalent to say that the direction vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ are linearly independent. Thus

$$\det[\mathbf{e}_1\mathbf{e}_2] = e_{11}e_{22} - e_{12}e_{21} > 0. \qquad (4.54)$$

Under this condition the boundary of the feasible region comprises all the points emanating from the origin in the two directions $\mathbf{g}_1 = (-e_{21},\, e_{11})^T$ and $\mathbf{g}_2 = (e_{22},\, -e_{12})^T$. For the minimum to be unique and at $\mathbf{y} = \mathbf{0}$ we must therefore have $\mathbf{c}^T\mathbf{g}_1 > 0$ and $\mathbf{c}^T\mathbf{g}_2 > 0$, that is

$$c_2e_{11} - c_1e_{21} > 0 \;\text{ and }\; c_1e_{22} - c_2e_{12} > 0. \qquad (4.55)$$

We therefore assume Inequality (4.55) is also satisfied.

We consider the transform $\mathbf{y} = \mathbf{A}\mathbf{x}$. (The LCNM method being a pure geometrical process with steps depending only on function values, is invariant to a linear transform of this sort.). It is readily verified that if

$$\mathbf{A} = \begin{bmatrix} k_1\mathbf{e}_1^T \\ k_2\mathbf{e}_2^T \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{l}_1 & \mathbf{l}_2 \end{bmatrix},$$

where $k_1 = (a_2 - a_1)(c_1e_{22} - c_2e_{12}) \,/\, [c(e_{11}e_{22} - e_{21}e_{12})]$ and $k_2 = (a_2 - a_1)$ $(c_2e_{11} - c_1e_{21}) \,/\, [c(e_{11}e_{22} - e_{21}e_{12})]$, then Problem 4.2 becomes

Problem 4.3

$$\min_{\mathbf{y}\in\mathbb{R}^2} \mathbf{c}^T\mathbf{y} = \min_{\mathbf{x}\in\mathbb{R}^2} \mathbf{c}^T\mathbf{A}\mathbf{x} = \min_{\mathbf{x}\in\mathbb{R}^2} c(x_1 + x_2),$$

subject to

$$\mathbf{e}_1^T\mathbf{y} = \mathbf{e}_1^T\mathbf{A}\mathbf{x} = k_1^{-1}\mathbf{l}_1^T\mathbf{x} \geq 0, \;\textit{and}\; \mathbf{e}_2^T\mathbf{y} = \mathbf{e}_2^T\mathbf{A}\mathbf{x} = k_2^{-1}\mathbf{l}_2^T\mathbf{x} \geq 0. \qquad (4.56)$$

From Inequalities (4.51), (4.54) and (4.55) we have that $k_1 > 0$ and $k_2 > 0$. The constraints $k_1^{-1}\mathbf{l}_1^T\mathbf{x} \geq 0$, and $k_2^{-1}\mathbf{l}_2^T\mathbf{x} \geq 0$ are therefore equivalent to

$$\mathbf{l}_1^T\mathbf{x} \geq 0, \;\text{ and }\; \mathbf{l}_2^T\mathbf{x} \geq 0.$$

Thus Problem 4.2 is identical to Problem 4.1.

To obtain the minimum, the LCNM method will have to form simplices that lie within the feasible cone and which progressively move towards the vertex origin. Figure 4.7 illustrates a possible sequence of such simplices $S^q = [\mathbf{x}_1^q : \mathbf{x}_2^q : \mathbf{x}_3^q]$ where all three vertices of every simplex are boundary vertices and the simplices each have two vertices on one boundary and one vertex on the other. However they 'flip' back and forth in the sense that the boundary with the two vertices alternates from simplex to simplex. We shall show that for the LCNM method there is always an initial simplex where this flipping continues indefinitely, so that the vertex is never reached in a finite number of steps.

4.4.2 The LCNM method for the LPP

Consider Problem 4.1, and suppose that we apply the LCNM method using as initial simplex S^0:

$$S^0 = [\mathbf{x}_1^0 : \mathbf{x}_2^0 : \mathbf{x}_3^0], \tag{4.57}$$

so that, as illustrated in Figure 4.7, $\mathbf{x}_1^0$ and $\mathbf{x}_3^0$ lie on the boundary of l_2 and $\mathbf{x}_2^0$ lies on the boundary of l_1. Then the sequence of simplices will have the alternating form illustrated in Figure 4.7 if application of the LCNM method at each simplex $S^q = [\mathbf{x}_1^q : \mathbf{x}_2^q : \mathbf{x}_3^q]$, leads to selection of the constrained reflection point $\mathbf{x}_{new}^q$ (as illustrated in Figure 4.6), as the next new point. Thus when alternating behaviour occurs the simplices satisfy the relation

$$S^{q+1} = [\mathbf{x}_1^{q+1} : \mathbf{x}_2^{q+1} : \mathbf{x}_3^{q+1}] = [\mathbf{x}_{new}^q : \mathbf{x}_1^q : \mathbf{x}_2^q], \quad q = 0, 1, 2, ... \tag{4.58}$$

This will occur if the following condition holds at each qth iteration.

Condition 4.1 (Relative values of $f(\mathbf{x})$)

$$f(\mathbf{x}_{new}^q) < f(\mathbf{x}_1^q) < f(\mathbf{x}_2^q) < f(\mathbf{x}_3^q) \quad q = 0, 1, 2....$$

Clearly, from the symmetry of Problem 4.1 this is equivalent to

Condition 4.2 (Distance of vertices $\mathbf{x}_i$)

$$||\mathbf{x}_{new}^q|| < ||\mathbf{x}_1^q|| < ||\mathbf{x}_2^q|| < ||\mathbf{x}_3^q|| \quad q = 0, 1, 2...$$

Thus, geometrically expressed, for Problem 4.1, alternating behaviour occurs if each move is a constrained reflection with the new point lying alternately on one constraint boundary and then on the other, and where each new point is nearer the origin than all previous simplex points. More explicitly we have the following:

Property 4.1 *Suppose the LCNM method is applied to Problem 4.1 using the initial simplex given by Equation (4.57). If Condition 4.2 holds, then the LCNM method generates the sequence of simplices given by Equation (4.58), with, at each iteration q, the new point lying alternately on l_1 and on l_2.*

Theorem 4.5 (on page 102) establishes the general behaviour of a sequence of simplices of the form given by Equation (4.58) whilst the Corollary 4.2 (on page 109) gives the condition on the initial simplex that ensures that Condition 4.2 is satisfied when Property 4.1 is obtained.

Our approach is to examine the *ratio of distances* of simplex points from the origin. We therefore have the following lemma.

Lemma 4.6 *Let k_{q+2} be*

$$k_{q+2} = \frac{||\mathbf{x}_2^q||}{||\mathbf{x}_1^q||} = \frac{\left|\left|\mathbf{x}_1^{q-1}\right|\right|}{\left|\left|\mathbf{x}_{new}^{q-1}\right|\right|} \quad \forall q = 1, 2, \ldots$$

the ratio of distances of the vertices of a simplex from the origin. Let $\mathbf{x}_1^0$, $\mathbf{x}_2^0$ and $\mathbf{x}_3^0$ be the three vertices of the initial simplex S^0 with $\mathbf{x}_1^0$, $\mathbf{x}_3^0$ on one line and $\mathbf{x}_2^0$ on the other. If we define the initial ratios of distances as

$$\frac{||\mathbf{x}_3^0||}{||\mathbf{x}_2^0||} = k_1, \quad \frac{||\mathbf{x}_2^0||}{||\mathbf{x}_1^0||} = k_2$$

Then for all iteration $q = 1, 2, \ldots,$

$$k_{q+2} = \frac{1}{k_{q+1}}\left(2 - \frac{1}{k_q k_{q+1}}\right) > 1 \tag{4.59}$$

Proof. Suppose that $||\mathbf{x}_1^0|| < ||\mathbf{x}_2^0|| < ||\mathbf{x}_3^0||$ (Condition 4.2). Then we have

$$k_1 > 1 \text{ and } k_2 > 1$$

and so

$$\frac{||x_3^0||}{||x_2^0||}\frac{||x_2^0||}{||x_1^0||} = \frac{||x_3^0||}{||x_1^0||} = k_1 k_2 > 1$$

The next candidate simplex is $[x_1^1, x_2^1, x_3^1] = [x_{new}^0, x_1^0, x_2^0]$. If the point x_{new}^0 is to lie on the same line as x_2^0 we must have

$$x_{new}^0 = \mu_0 x_2^0$$

and if this candidate simplex is formed from the LCNM algorithm then straightforward geometrical consideration (See Appendix C in Section C.1) that we must have

$$\mu_0 = \frac{1}{2 - \frac{1}{k_1 \cdot k_2}}.$$

Moreover if this new point is to be nearer to the origin than all points of the initial simplex we must have the next ratio, k_3, satisfying

$$k_3 = \frac{||x_2^1||}{||x_1^1||} = \frac{||x_1^0||}{||x_{new}^0||} = \frac{||x_1^0||}{\mu_0\, ||x_2^0||} = \frac{1}{\mu_0 k_2} > 1$$

Namely,

$$k_3 = \frac{1}{k_2}\left(2 - \frac{1}{k_1 k_2}\right) > 1$$

We see therefore that successive ratios satisfy the recursion

$$k_{q+2} = \frac{1}{k_{q+1}}\left(2 - \frac{1}{k_q k_{q+1}}\right) > 1 \ \ \forall q = 1, 2, \ldots$$

and the LCNM algorithm will produce the required sequence of simplices provided

$$k_m > 1 \ \ \forall m = 0, 1, \ldots \ . \tag{4.60}$$

■

Theorem 4.5 *Consider Problem 4.1. If the LCNM algorithm is applied to Problem 4.1 beginning with the simplex given by Equation (4.57), let S^q be the alternating sequence as defined in Equation (4.58) and let the two initial ratios be*

$$k_1 = a \ \ \wedge \ \ k_2 = b$$

Then for all $n = 1, 2, \ldots,$

$$k_{2n+1} = \frac{\|\mathbf{x}_3^{2n}\|}{\|\mathbf{x}_2^{2n}\|} = \frac{1}{ab^2}(2nab - (2n-1)) \prod_{i=1}^{n-1} \left[\frac{2iab - (2i-1)}{(2i+1)ab - 2i}\right]^2, \tag{4.61}$$

and

$$k_{2n} = \frac{\|\mathbf{x}_2^{2n}\|}{\|\mathbf{x}_1^{2n}\|} = \frac{(2n-1)ab - 2(n-1)}{a} \prod_{i=1}^{n-1} \left[\frac{(2i-1)ab - (2i-2)}{2iab - (2i-1)}\right]^2, \tag{4.62}$$

where $\prod_{i=1}^{0} [\cdot] = 1.$

Moreover

1) k_{2n+1} *is a decreasing sequence as* n *increases with*

$$\frac{k_{2(n+1)+1}}{k_{2n+1}} = 1 - \frac{w^2}{[(2nw+1)+w]^2} \quad \forall w > 0 \wedge n = 1, 2, \ldots$$

2)

$$\lim_{n\to\infty} k_{2n+1} = 2wa \frac{\Gamma^2\left(\frac{1+w}{2w}\right)}{\Gamma^2\left(\frac{1}{2w}\right)} \quad \forall w > 0$$

3) k_{2n} *is a decreasing sequence as* n *increases with*

$$\frac{k_{2(n+1)}}{k_{2n}} = 1 - \frac{w^2}{(2nw+1)^2} \quad \forall w > 0 \wedge n = 1, 2, \ldots$$

4)

$$\lim_{n\to\infty} k_{2n} = \frac{1}{2wa} \frac{\Gamma^2\left(\frac{1}{2w}\right)}{\Gamma^2\left(\frac{1+w}{2w}\right)} \quad \forall w > 0.$$

where $w = ab - 1$

Proof. Suppose the two initial ratios are

$$k_1 = a \quad \wedge \quad k_2 = b$$

From Lemma 4.6 we evaluate Equation (4.59) for $q = 1, 2, 3, \ldots$ we obtain

$$k_3 = \frac{1}{k_2}\left(2 - \frac{1}{k_1 k_2}\right) = \frac{1}{ab^2}(2ab - 1)$$

$$k_4 = \frac{1}{k_3}\left(2 - \frac{1}{k_2 k_3}\right) = \frac{(3ab-2)}{a}\frac{(ab)^2}{(2ab-1)^2}$$

$$k_5 = \frac{1}{k_4}\left(2 - \frac{1}{k_3 k_4}\right) = \frac{1}{ab^2}\frac{(4ab-3)(2ab-1)^2}{(3ab-2)^2}$$

$$k_6 = \frac{1}{k_5}\left(2 - \frac{1}{k_4 k_5}\right) = \frac{(5ab-4)}{a}\frac{(3ab-2)^2(ab)^2}{(4ab-3)^2(2ab-1)^2}$$

$$k_7 = \frac{1}{k_6}\left(2 - \frac{1}{k_5 k_6}\right) = \frac{1}{ab^2}\frac{(6ab-5)(4ab-3)^2(2ab-1)^2}{(5ab-4)^2(3ab-2)^2}$$

$$k_8 = \frac{1}{k_7}\left(2 - \frac{1}{k_6 k_7}\right) = \frac{(7ab-6)}{a}\frac{(5ab-4)^2(3ab-2)^2(ab)^2}{(6ab-5)^2(4ab-3)^2(2ab-1)^2}$$

$$k_9 = \frac{1}{k_8}\left(2 - \frac{1}{k_7 k_8}\right) = \frac{1}{ab^2}\frac{(8ab-7)(6ab-5)^2(4ab-3)^2(2ab-1)^2}{(7ab-6)^2(5ab-4)^2(3ab-2)^2}$$

By examining of k_m we therefore obtain that for all odd mth cases

$$k_{2n+1} = \frac{\|\mathbf{x}_3^{2n}\|}{\|\mathbf{x}_2^{2n}\|} = \frac{1}{ab^2}(2nab - (2n-1))\prod_{i=1}^{n-1}\left[\frac{2iab-(2i-1)}{(2i+1)ab-2i}\right]^2$$

On the other hand, for all even mth cases, we have

$$k_{2n} = \frac{\|\mathbf{x}_2^{2n}\|}{\|\mathbf{x}_1^{2n}\|} = \frac{(2n-1)ab - 2(n-1)}{a}\prod_{i=1}^{n-1}\left[\frac{(2i-1)ab-(2i-2)}{2iab-(2i-1)}\right]^2$$

Part 1.

From Equation (4.61), we have

$$k_{2n+1} = \frac{1}{ab^2}(2nab - (2n-1))\prod_{i=1}^{n-1}\left[\frac{2iab-(2i-1)}{(2i+1)ab-2i}\right]^2 \quad \forall n = 1, 2, \ldots$$

If $w = ab - 1$, then we obtain

$$k_{2n+1} = \frac{1}{b}\frac{2nw+1}{w+1}\prod_{i=1}^{n-1}\left[\frac{2iw+1}{2iw+w+1}\right]^2 \quad \forall w > 0 \wedge n = 1, 2, \dots . \tag{4.63}$$

If k_{2n+1} is a decreasing sequence as n increases for all $w > 0$, then

$$k_{2n+1} > k_{2(n+1)+1} \quad \forall w > 0 \wedge n = 1, 2, \dots .$$

Using Equation (4.63) for n and $n+1$, we have

$$\frac{k_{2(n+1)+1}}{k_{2n+1}} = \frac{\frac{1}{b}\frac{2(n+1)w+1}{w+1}\prod_{i=1}^{n}\left[\frac{2iw+1}{(2i+1)w+1}\right]^2}{\frac{1}{b}\frac{2nw+1}{w+1}\prod_{i=1}^{n-1}\left[\frac{2iw+1}{(2i+1)w+1}\right]^2} = \frac{[2(n+1)w+1]\left[\frac{2nw+1}{(2n+1)w+1}\right]^2}{2nw+1}$$

$$\frac{k_{2(n+1)+1}}{k_{2n+1}} = \frac{[2(n+1)w+1]}{2nw+1}\cdot\frac{(2nw+1)^2}{[(2n+1)w+1]^2} = \frac{(2nw+1+2w)(2nw+1)}{(2nw+1+w)^2}$$

$$\frac{k_{2(n+1)+1}}{k_{2n+1}} = \frac{(2nw+1)^2+2w(2nw+1)}{(2nw+1)^2+2w(2nw+1)+w^2} = 1 - \frac{w^2}{[(2nw+1)+w]^2}$$

$$\frac{k_{2(n+1)+1}}{k_{2n+1}} = 1 - \frac{w^2}{[(2nw+1)+w]^2} \quad \forall w > 0 \wedge n = 1, 2, \dots \tag{4.64}$$

Since $0 < \frac{k_{2(n+1)+1}}{k_{2n+1}} < 1$ for all $w > 0$ and $n = 1, 2, \dots$ (see Equation (4.64)), then clearly $k_{2n+1} > k_{2n+3}$ for all $w > 0$ and $n = 1, 2, \dots$. This is, k_{2n+1} is a decreasing sequence as n increases.

Part 2.

From Equation (4.63) we define $g_0(n)$ as

$$g_0(n) = \prod_{i=1}^{n-1}\left[\frac{2iw+1}{2iw+w+1}\right]^2 = \frac{\prod_{i=1}^{n-1}\left[1+\frac{1/2w}{i}\right]^2}{\prod_{i=1}^{n-1}\left[1+\frac{(1+w)/2w}{i}\right]^2} \quad \forall w > 0 \quad \wedge \quad n = 1, 2, \dots . \tag{4.65}$$

Using Equation (C.12) of Appendix C (Section C.2) and Equation (4.63), we have

$$k_{2n+1} = a\frac{(2nw+1)}{(n-1)}\frac{\Gamma^2_{n-1}\left(\frac{1+w}{2w}\right)}{\Gamma^2_{n-1}\left(\frac{1}{2w}\right)} \quad \forall w > 0 \quad \wedge \quad n = 1, 2, \dots \ . \tag{4.66}$$

Let K_o be $\lim_{n\to\infty} k_{2n+1}$

$$K_o = 2wa\frac{\Gamma^2\left(\frac{1+w}{2w}\right)}{\Gamma^2\left(\frac{1}{2w}\right)} \quad \forall w > 0 \ . \tag{4.67}$$

Part 3.

From Equation (4.62), we obtain

$$k_{2n} = \frac{(2n-1)ab - 2(n-1)}{a}\prod_{i=1}^{n-1}\left[\frac{(2i-1)ab-(2i-2)}{2iab-(2i-1)}\right]^2 \quad \forall n = 1, 2, \dots$$

Using $w = ab - 1$, we have

$$k_{2n} = \frac{(2n-1)w+1}{a}\prod_{i=1}^{n-1}\left[\frac{(2i-1)w+1}{2iw+1}\right]^2 \quad \forall n = 1, 2, \dots \ . \tag{4.68}$$

k_{2n} is a decreasing sequence as n increases.

Here we will prove that k_{2n} is a decreasing sequence as n increases for all $w > 0$, that is,

$$k_{2n} > k_{2(n+1)} \quad \forall w > 0 \wedge n = 1, 2, \dots \ .$$

Using Equation (4.68) for n and $n+1$, we have

$$\frac{k_{2(n+1)}}{k_{2n}} = \frac{\frac{(2(n+1)-1)w+1}{a}\prod_{i=1}^{n}\left[\frac{(2i-1)w+1}{2iw+1}\right]^2}{\frac{(2n-1)w+1}{a}\prod_{i=1}^{n-1}\left[\frac{(2i-1)w+1}{2iw+1}\right]^2} = \frac{\frac{(2(n+1)-1)w+1}{a}\left[\frac{(2n-1)w+1}{2nw+1}\right]^2\prod_{i=1}^{n-1}\left[\frac{(2i-1)w+1}{2iw+1}\right]^2}{\frac{(2n-1)w+1}{a}\prod_{i=1}^{n-1}\left[\frac{(2i-1)w+1}{2iw+1}\right]^2}$$

$$\frac{k_{2(n+1)}}{k_{2n}} = \frac{\frac{(2(n+1)-1)w+1}{a}\left[\frac{(2n-1)w+1}{2nw+1}\right]^2}{\frac{(2n-1)w+1}{a}}$$

$$\frac{k_{2(n+1)}}{k_{2n}} = \frac{[(2(n+1)-1)w+1]}{[(2n-1)w+1]}\cdot\left[\frac{(2n-1)w+1}{2nw+1}\right]^2$$

$$\frac{k_{2(n+1)}}{k_{2n}} = \frac{(2nw+1+w)}{1} \cdot \frac{(2nw+1-w)}{[2nw+1]^2} = \frac{(2nw+1)^2-w^2}{(2nw+1)^2}$$

$$\frac{k_{2(n+1)}}{k_{2n}} = 1 - \frac{w^2}{(2nw+1)^2} \quad \forall w > 0 \wedge n = 1, 2, \ldots \tag{4.69}$$

Since $0 < \frac{k_{2(n+1)}}{k_{2n}} < 1$ (see Equation (4.69)), then clearly $k_{2n} > k_{2n+2}$ for all $w > 0$ and $n = 1, 2, \ldots$. This is, k_{2n} is a decreasing sequence as n increases.

Part 4.

For $w \neq 1$ and $w > 0$

From Equation (4.68) let $g_e(n)$ be

$$g_e(n) = \prod_{i=1}^{n-1} \left[\frac{2iw+1-w}{2iw+1}\right]^2 = \frac{\prod_{i=1}^{n-1}\left[1+\frac{(1-w)/2w}{i}\right]^2}{\prod_{i=1}^{n-1}\left[1+\frac{1/2w}{i}\right]^2} \quad \forall w \neq 1 \ \wedge w > 0 \ \wedge \ n = 1, 2, \ldots . \tag{4.70}$$

Plugging Equation (C.13) of Appendix C (Section C.2) into Equation (4.68), we have

$$k_{2n} = \frac{1}{a}\frac{(2n-1)w+1}{(n-1)(1-w)^2}\frac{\Gamma_{n-1}^2\left(\frac{1}{2w}\right)}{\Gamma_{n-1}^2\left(\frac{1-w}{2w}\right)} \quad \forall w \neq 1 \ \wedge w > 0 \ \wedge \ n = 1, 2, \ldots . \tag{4.71}$$

Letting $n \to \infty$

$$\lim_{n\to\infty} k_{2n} = K_e = \frac{2w}{a(1-w)^2}\frac{\Gamma^2\left(\frac{1}{2w}\right)}{\Gamma^2\left(\frac{1-w}{2w}\right)} \quad \forall w \neq 1 \ \wedge w > 0 \tag{4.72}$$

For $w = 1$

$$k_{2n} = \frac{2n}{a}\prod_{i=1}^{n-1}\left[\frac{2i}{2i+1}\right]^2 = \frac{2n}{a}\frac{1}{\prod_{i=1}^{n-1}\left[1+\frac{1/2}{i}\right]^2} \quad \forall w = 1 \ \wedge \ n = 1, 2, \ldots . \tag{4.73}$$

Let $P_4(n)$ be

$$P_4(n) = \prod_{i=1}^{n-1}\left[1+\frac{1/2}{i}\right]^2 = \frac{n-1}{\frac{1}{4}\Gamma_{n-1}^2\left(\frac{1}{2}\right)} = \frac{4(n-1)}{\Gamma_{n-1}^2\left(\frac{1}{2}\right)} \quad \forall w = 1 \ \wedge \ n = 1, 2, \ldots . \tag{4.74}$$

Using Equation (4.74) in Equation (4.73), we obtain

$$k_{2n} = \frac{2n}{a}\frac{1}{\frac{4(n-1)}{\Gamma^2_{n-1}(\frac{1}{2})}} = \frac{n}{2a(n-1)}\Gamma^2_{n-1}(1/2) \;\; \forall w = 1 \;\wedge\;\; n = 1, 2, \ldots\;.$$

Letting $n \to \infty$, we have

$$K_e = \frac{\Gamma^2(1/2)}{2a} = \frac{\pi}{2a} \quad \forall w = 1 \tag{4.75}$$

Convergence value of K_e

From Equations (4.72) and (4.75), we rewrite

$$K_e = \begin{cases} \frac{2w}{a(1-w)^2}\frac{\Gamma^2(\frac{1}{2w})}{\Gamma^2(\frac{1-w}{2w})} & \text{if } \; w \neq 1 \;\wedge w > 0 \\ \frac{\pi}{2a} & \text{if } \; w = 1 \end{cases} \tag{4.76}$$

From the called *fundamental recurrence formula*

$$\Gamma(z+1) = z\Gamma(z) \tag{4.77}$$

We shall rewrite Equation (4.76) for $w \neq 1$ and $w > 0$ using the following change of variable

$$z = \frac{1-w}{2w}, \;\; \text{so } z+1 = \frac{1+w}{2w} \tag{4.78}$$

Substituting Equation (4.78) into Equation (4.77),

$$\Gamma\left(\frac{1+w}{2w}\right) = \frac{1-w}{2w}\Gamma\left(\frac{1-w}{2w}\right) \tag{4.79}$$

Using Equation (4.79) in Equation (4.76), we obtain for $w \neq 1$ and $w > 0$

$$K_e = \frac{2w}{a(1-w)^2}\frac{\Gamma^2\left(\frac{1}{2w}\right)}{\Gamma^2\left(\frac{1-w}{2w}\right)} = \frac{2w}{a(1-w)^2}\frac{\Gamma^2\left(\frac{1}{2w}\right)}{\frac{\Gamma^2(\frac{1+w}{2w})}{(\frac{1-w}{2w})^2}}$$

$$K_e = \frac{1}{2wa}\frac{\Gamma^2\left(\frac{1}{2w}\right)}{\Gamma^2\left(\frac{1+w}{2w}\right)} \tag{4.80}$$

Therefore, K_e may be rewritten as

$$K_e = \begin{cases} \frac{1}{2wa}\frac{\Gamma^2(\frac{1}{2w})}{\Gamma^2(\frac{1+w}{2w})} & \text{if } w \neq 1 \wedge w > 0 \\ \frac{\pi}{2a} & \text{if } w = 1 \end{cases}, \tag{4.81}$$

which can be rewritten as

$$K_e = \frac{1}{2wa}\frac{\Gamma^2\left(\frac{1}{2w}\right)}{\Gamma^2\left(\frac{1+w}{2w}\right)} \quad \forall w > 0. \tag{4.82}$$

∎

From Equations (4.67) and (4.82), we have

$$K_o K_e = 1 \quad \forall w > 0 \tag{4.83}$$

The implication of this is as follows.

Corollary 4.2 *For any initial simplex S^0 for which a, b satisfy $K_o = 1$, the LCNM method will give rise to an unending alternating sequence of simplices.*

Proof. From Equation (4.83) we see that if a and b satisfy $K_o = 1$ then this automatically gives $K_e = 1$. As the Theorem shows that the k_{2n+1} and k_{2n} are decreasing sequences tending to K_o and K_e respectively, this implies that all the $k_m > 1$, $m = 0, 1, 2..$ so that Inequality (4.60) is satisfied and the corollary follows. ∎

Though $K_o = 1$ gives the relation between a and b implicitly, pairs are easily found numerically by setting a, say, and then finding b iteratively to satisfy $K_o = 1$. In fact an elementary analysis of the form of K_o shows that if $a = 1 + \delta$, $\delta > 0$, then $b = 1 + \delta - 2\delta^2 + \mathcal{O}(\delta^3)$.

4.4.3 Analysis of the rate of convergence

In this subsection we shall analyse the rate of convergence of $\mathbf{x}_1^{2n}$ through its distance from the origin, when any collapsed simplex does not take place during the application of the LCNM method.

$$k_{2n+1}k_{2n} = \frac{\left\|\mathbf{x}_3^{2n}\right\|}{\left\|\mathbf{x}_2^{2n}\right\|}\frac{\left\|\mathbf{x}_2^{2n}\right\|}{\left\|\mathbf{x}_1^{2n}\right\|} = \frac{\left\|\mathbf{x}_3^{2n}\right\|}{\left\|\mathbf{x}_1^{2n}\right\|} > 1 \quad \forall n = 0, 1, \ldots$$

Since $\mathbf{x}_3^{2n} = \mathbf{x}_1^{2n-2}$ for all $n = 1, 2, \ldots$, then

$$k_{2n+1}k_{2n} = \frac{\left\|\mathbf{x}_1^{2n-2}\right\|}{\left\|\mathbf{x}_1^{2n}\right\|},$$

therefore

$$d(2n) = \frac{1}{k_{2n+1} \cdot k_{2n}} d(2n-2) \quad \forall n = 1, 2, \ldots \tag{4.84}$$

where $d(2n)$ is $\left\|\mathbf{x}_1^{2n}\right\|$

From Equations (4.63) and (4.68), we obtain

$$k_{2n+1}k_{2n} = \frac{1}{ab} \frac{(2nw+1)[(2n-1)w+1]}{w+1} \prod_{i=1}^{n-1} \left[\frac{(2i-1)w+1}{(2i+1)w+1}\right]^2 \tag{4.85}$$

Using $w = ab - 1$ and developing the product, we have

$$k_{2n+1}k_{2n} = \frac{(2nw+1)[(2n-1)w+1]}{(w+1)^2} \cdot \left[\frac{w+1}{(2n-1)w+1}\right]^2$$

which can be rewritten as

$$k_{2n+1}k_{2n} = \frac{(2nw+1)}{(2n-1)w+1} > 1 \quad \forall w > 0 \quad \wedge \quad n = 2, 3, \ldots.$$

Therefore

$$\frac{1}{k_{2n+1}k_{2n}} = \frac{2nw+1-w}{2nw+1} = 1 - \frac{w}{2nw+1} \tag{4.86}$$

Using Equation (4.86) in Equation (4.84), we obtain the recursion

$$d(2n) = \left[\frac{(2n-1)w+1}{2nw+1}\right] d(2n-2) \quad \forall n = 1, 2, \ldots \tag{4.87}$$

From Equation (4.87), we have

$$d(2) = \left[\frac{w+1}{2w+1}\right] d(0)$$

$$d(4) = \left[\frac{3w+1}{4w+1}\right] d(2) = \left[\frac{3w+1}{4w+1}\right] \left[\frac{w+1}{2w+1}\right] d(0)$$

$$d(6) = \left[\frac{5w+1}{6w+1}\right] d(4) = \left[\frac{5w+1}{6w+1}\right] \left[\frac{3w+1}{4w+1}\right] \left[\frac{w+1}{2w+1}\right] d(0)$$

As a result of this, we write

$$d(2n) = d(0) \prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{2iw+1}\right] \quad \forall w > 0 \quad \wedge \quad n = 1, 2, \ldots, \tag{4.88}$$

where $d(0) = \|\mathbf{x}_1^0\|$.

Rate of convergence

From Equation (4.88), we obtain

$$d^2(2n) = d^2(0) \prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{2iw+1}\right]^2 \tag{4.89}$$

Now, if $y > x > w > 0$, then

$$\frac{x}{y} < \frac{x+w}{y+w}, \tag{4.90}$$

and

$$\frac{x}{y} > \frac{x-w}{y-w}. \tag{4.91}$$

Using Inequality (4.90) in Equation (4.89), we have for all $n = 1, 2, \ldots,$

$$d^2(2n) < d^2(0) \prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{2iw+1}\right] \left[\frac{(2i-1)w+1+w}{2iw+1+w}\right]$$

$$d^2(2n) < d^2(0) \prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{(2i+1)w+1}\right] \tag{4.92}$$

Since,

$$\prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{(2i+1)w+1}\right] = \frac{w+1}{(2n+1)w+1}, \tag{4.93}$$

then plugging Equation (4.93) into Inequality (4.92), we obtain

$$d^2(2n) < d^2(0) \left[\frac{w+1}{(2n+1)w+1}\right]. \tag{4.94}$$

Using Inequality (4.91) in Equation (4.89), we have for all $n = 1, 2, \ldots,$

$$d^2(2n) > d^2(0) \prod_{i=1}^{n} \left[\frac{(2i-1)w+1}{2iw+1} \right] \left[\frac{(2i-1)w+1-w}{2iw+1-w} \right]$$

$$d^2(2n) > d^2(0) \prod_{i=1}^{n} \left[\frac{(2i-2)w+1}{2iw+1} \right] \tag{4.95}$$

We have

$$\prod_{i=1}^{n} \left[\frac{(2i-2)w+1}{2iw+1} \right] = \frac{1}{2nw+1} \tag{4.96}$$

From Inequality (4.95) and Equation (4.96), we obtain

$$d^2(2n) > d^2(0) \left[\frac{1}{2nw+1} \right] \tag{4.97}$$

In consequence, from Inequalities (4.94) and (4.97)

$$d(0) \left[\frac{1}{2nw+1} \right]^{1/2} < d(2n) < d(0) \left[\frac{w+1}{(2n+1)w+1} \right]^{1/2} \tag{4.98}$$

This result shows that, for large q, $d(q) = \mathcal{O}(q^{-1/2})$. This rate of convergence is slow and is comparable only with rates of convergence of stochastic estimators rather than for deterministic situations.

4.5 Conclusions

A theoretical study of the behaviour of the LCNM algorithm has been carried out to understand some of its behaviour. We have examined in detail the convergence rate of the LCNM algorithm only for particular situations so it does not correspond to a rigorous study of convergence of the LCNM algorithm. However, it allows us to appreciate some of the behaviour of the method when the LCNM algorithm.

The class of convex functions described by McKinnon (1998) is one where problems may be encountered. Therefore, we have to say that the LCNM algorithm may not converge for all convex function. Nevertheless, the analysis of convergence presented in this chapter tries to find some features of the LCNM algorithm, when the algorithm is approaching to a local minimum.

The symmetrically constrained linear optimization case studied in Section 4.4 permitted demonstrating how the LCNM algorithm becomes exceptionally slow, even in the very simple situation of two symmetrically constrained linear optimization where the rate of convergence is of order $\mathcal{O}(q^{-1/2})$. This fact and the non guarantee of convergence of the NM algorithm when the objective function belongs to the class of functions described by McKinnon (1998) induce us to see the LCNM algorithm as a non-rigorous method. However, the variations of the NM method developed by Barton and Ivey (1996) and Humphrey and Wilson (2000) indicate a good enough practical performance, when the objective function presents noise. We believe therefore that the apparent theoretical reservations do not therefore rule out the potential practical usefulness of the LCNM method.

Chapter 5

A modified LCNM method

5.1 Introduction

In this chapter we consider the possibility of eliminating the potential slow convergence of the LCNM method by inducing an early collapse of the simplex, once a constraint is encountered.

One danger is that when a collapse of the simplex occurs prematurely, the LCNM algorithm orientates the search of the optimum on the boundary of the feasible region. Thus if the global minimum of the objective function were an interior point, this new procedure of inducing a collapse of the simplex could eventually miss it.

Because of this we implemented a procedure that prematurely collapses the simplex only during the first stage of the LCNM algorithm. This avoids the method repeatedly trying to collapse the simplex in other stages and it prevents the search being confined to a boundary what the true minimum is an interior point.

The structure of this chapter is as follows. In Section 5.2 we present the modified LCNM algorithm using a new criterion for inducing the collapse of the simplex prematurely. A set of preliminary experiments is shown in Section 5.3 for studying the behaviour of the modified method in comparison with the original version. Results of a large set of test problems are reported in Section 5.4 for comparing the LCNM algorithm, its premature collapse mode and the Subrahmanyam method (SM). Finally, conclusions of the modified LCNM algorithm and its original version are presented in Section 5.5.

5.2 Modification of the LCNM method

The study of convergence of the LCNM algorithm for the symmetrically linear constrained linear optimization problem helped us to detect a potential trouble, when the constrained global optimum is located on the boundary and a collapse of the simplex does not take place during the application of the algorithm. The algorithm could thereby become expensive in terms of the number of function evaluations. In fact, numerical examples have evidenced this behaviour when the objective function is approximately linear in the feasible region, even in linearly constrained convex quadratic optimization problems, where the collapse of the simplex occurs only after a considerable number of iterations.

We have therefore studied a variant of the LCNM algorithm using a criterion that induces the collapse of the current non-collapsed simplex in premature manner by a procedure, called Premature Collapse Procedure (PCP).

We call the method Linear Constrained Nelder-Mead method with Premature Collapse (LCNM+PC) algorithm.

The method employs the PCP only during the first stage of the LCNM+PC algorithm. The number of vertices of the current simplex that is on each linear constraint boundary is monitored. So if the current qth simplex of v_q vertices has $v_q - 1$ vertices on any linear constraint boundary, the PCP identifies the vertex that is not on the boundary, replacing it by the centroid of the remaining hyperface $H_v^{[q]}$. Thus in the next iteration, a collapse of the simplex is initiated.

The LCNM+PC algorithm therefore applies the procedure of reduction of vertices in the same way as is performed by the original version. However this is only applied during the first stage of the LCNM+PC algorithm but not in any succeeding stage.

Once of a point $\mathbf{x}_{\min}^{[1]}$ is found through convergence of the algorithm, then its coordinates are employed for starting the following stage, building a new simplex in either the sub-space where the previous convergent point $\mathbf{x}_{\min}^{[(s\text{-}1)\text{th}]}$ lies or in the entire feasible region.

All subsequent stages employ the original LCNM method.

It is worthwhile mentioning that the modified algorithm employed the same stopping rule as the original version for all stages. Nevertheless, the stopping tolerance is relaxed during the first stage, because the algorithm does not need a high accuracy for obtaining the point $\mathbf{x}_{\min}^{[1]}$. Thus reduces the number of function evaluations.

To include this approach to the LCNM algorithm, modifications at the steps 10 and 11 of the LCNM algorithm were required, and they are summarized as follows.

5.2.1 Pseudocode of the modification

The modification of the LCNM algorithm is here shown by including some changes in the steps 10 and 11 of the original algorithm.

Step 10. Test of termination criterion I

Sort $f_j \quad \forall \quad j = 1, 2, \ldots, v$ for determining $\mathbf{x}_{\min}$.

Compute the following inequality for the current simplex according to the stage

Given $flag_{\text{modepc}} \in \{true, false\}$

Select the case according to $flag_{\text{modepc}}$

Case *true*:

In this case, the LCNM algorithm is in the first stage

and for a fixed $\eta > 0$, verify $\max\limits_{\mathbf{x}_j} \|\mathbf{x}_j - \mathbf{x}_{\min}\| \leq \eta$

Case *false*:

In this case, the LCNM algorithm is in the sth ($s > 1$) stage,

verify

$$\max_{\mathbf{x}_j} \|\mathbf{x}_j - \mathbf{x}_{\min}\| \leq \begin{cases} \eta_1 \|\mathbf{x}_{\min}\| & \text{if } \|\mathbf{x}_{\min}\| > \varepsilon, \\ \eta_2 & \text{otherwise,} \end{cases}$$

where ε is a very small positive number.

End select

if one of above inequalities is satisfied

then go to step 13, for keeping the optimum solution at $(s+1)$th stage

and testing the stopping rule

else go to step 11.

The above step seeks an initial local optimum to the level of tolerance η, where η could be fixed to a high level, so it is approached a first local optimum.

Step 11. Test for estimating a new active constraint or increasing of r

$\Delta r \leftarrow 0$

Do, for all $i = 1, 2, \ldots, k \wedge i \notin S_{active}$

if $a_i^T[\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_v] - b_i \mathbf{1}_v^T = \mathbf{0}_v^T$

then $\Delta r \leftarrow \Delta r + 1$ and identify the current linear constraint $l_i(\mathbf{x})$,

as active constraint. Thus, $S_{active} \leftarrow S_{active} \cup \{i\}$

End do

if $\Delta r > 0$ **then** go to step 12.

else

if the LCNM algorithm is in the first stage

then perform **Premature collapse procedure** and

go to step 5 of the LCNM algorithm.

else go to step 5 of the LCNM algorithm

Premature Collapse Procedure

This procedure verifies whether the current simplex meets the condition for inducing its collapse prematurely or not. When the current simplex does satisfy the conditions, the procedure induces the collapse of the simplex at step 2.

Algorithm 5.1 (Premature Collapse Procedure) *Start procedure*

Step 1. Test for determining if a premature collapse of the current simplex can take place

$\Delta r \leftarrow 0$

$v_{l_i} \leftarrow 0 \quad \forall i = 1, \ldots k \wedge i \notin S_{active}$

Estimate the number of vertices v_{l_i} that are on the boundary

of each $l_i \notin S_{active}$ using the following procedure,

Do, *for all $i = 1, 2, \ldots, k \wedge i \notin S_{active}$*

Do, *for all $j = 1, 2, \ldots, v$*

if $a_i^T \mathbf{x}_j - b_i = 0$ ***then*** $v_{l_i} \leftarrow v_{l_i} + 1$

End do

End do

Estimate the maximum v_{l_i} $\forall i = 1, \ldots k \wedge i \notin S_{active}$ *and denote it as* $M_{vertices}$,

and keep its subscript i *as* pc

if $M_{vertices} = v - 1$

then *go to step 2 for inducing a collapse to the current simplex onto the* l_{pc} *constraint.*

else *return.*

Step 2. Become the current simplex into collapsed one on the l_{pc} ***constraint***

Because there only exists a j*th vertex of the current simplex that produces a positive residual* $\breve{r}_{pc}(\mathbf{x}_j) = a_{pc}^T \mathbf{x}_j - b_{pc}$ *to the pcth constraint, it must be identified for replacing it by the centroid of the vertices that are on the boundary of the* l_{pc} *constraint.*

Let $\mathbf{x}_{in}$ *denote the vertex that produces a positive residual.*

$\mathbf{x}_{cum} = \mathbf{0}_d$

Do, *for all* $j = 1, 2, \ldots, v$

if $a_{pc}^T \mathbf{x}_j - b_{pc} = 0$ ***then*** $\mathbf{x}_{cum} \leftarrow \mathbf{x}_{cum} + \mathbf{x}_j$ ***else*** $in \leftarrow j$

End do

$\mathbf{x}_{in} \leftarrow \frac{1}{v-1}\mathbf{x}_{cum}$ *for obtaining a collapsed simplex*

Return

End procedure

5.3 Experiments

With the aim of establishing the potentiality of the new algorithm, we compared the LCNM+PC algorithm, the LCNM algorithm and the SM through some preliminary numerical examples.

5.3.1 Experiment 1

Quadratic objective function with global optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

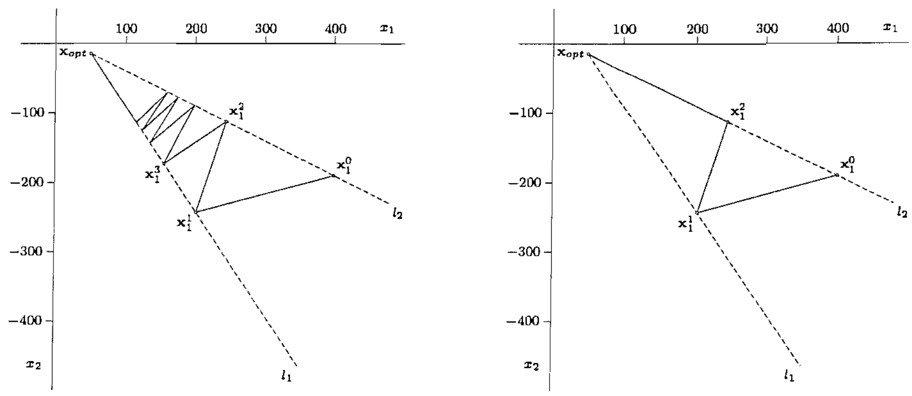
subject to:

$$\begin{aligned} l_1:&\quad 3x_1 + 2x_2 \;\geq\; 120 \\ l_2:&\quad x_1 + 2x_2 \;\leq\; 20 \end{aligned}$$

The parameter d represents the dimension of the Euclidean space and the constrained global optimum solution is given by $\mathbf{x}_{\min} = (50, -15, \underbrace{0, 0, \ldots, 0}_{(d-2)\text{ times}})^T$ with $f(\mathbf{x}_{\min}) = 2725$.

For each scenario, the initial point of the simplex was $\mathbf{x}_{initial} = (400, -400, \underbrace{400, \ldots, 400}_{(d-2)\text{ times}})^T$, and the stopping parameters $\eta = 0.1$, $\eta_1 = \eta_2 = 10^{-6}$ and $\Delta = 10^{-5}$.

Figure 5.1 depicts the trajectory of the minimum vertex $\mathbf{x}_1^q$ of each qth simplex for the case of two dimensions and both versions of the LCNM algorithm. The dashed lines are the boundaries of the linear constraints, whilst the solid line segments represent the path of the vertex $\mathbf{x}_1^q$.



(a) The LCNM algorithm

(b) The LCNM+PC algorithm

Figure 5.1: Path of the minimum vertex $\mathbf{x}_1^q$ of the simplices for the quadratic objective function with global minimum on the boundary of the region.

From Figure 5.1(a), the minimum vertex $\mathbf{x}_1^q$ of each qth simplex searches both boundaries ten times, so the method searches the minimum with more evaluations than the case shown in Figure 5.1(b), where the LCNM+PC algorithm was applied.

Table 5.1: Summary of Experiment 1.

	d	2	3	4	5	6	7	8
LCNM	NE	36	209	334	509	2554	2725*	4518
	DTP	7.8E-13	3.75E-6	1.01E-5	1.84E-5	1.84E-5	1.66E-5	2.39E-5
LCNM +PC	NE	19	157	816	553	632	1103	3800
	DTP	6.44E-14	1.25E-5	1.09E-5	1.76E-5	1.66E-5	1.97E-5	2.76E-3
SM	NE	330	∞	∞	∞	∞	∞	∞
	DTP	5.05E-5						

Table 5.1 shows results for each dimension d when the LCNM algorithm, the LCNM+PC algorithm and the SM were applied to the problem. We must point out that for the case $d = 7$ the coefficient α of the LCNM algorithm was fitted to 0.96 to obtain the best performance. As can also be seen from the table, the SM only converges for the case when the dimension is equal to two.

5.3.2 Experiment 2

Quadratic objective function when the global optimum point is an interior point.

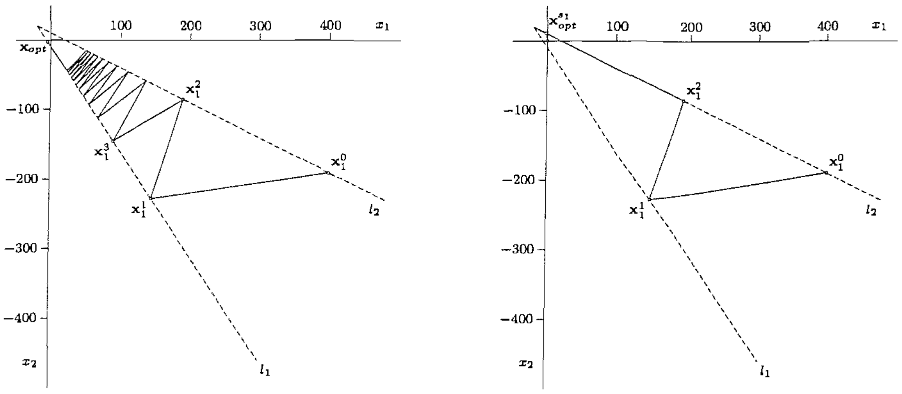
$$\min_{\mathrm{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to:

$$\begin{aligned} l_1\colon\ 3x_1 + 2x_2 &\geq -20 \\ l_2\colon\ x_1 + 2x_2 &\leq 20 \end{aligned}$$

The parameter d represents the dimension of the Euclidean space and the local optimum solution is at the origin.

The algorithms were performed with stopping parameters $\eta = 0.1$, $\eta_1 = \eta_2 = 10^{-6}$ and $\Delta = 10^{-5}$, an initial point $\mathrm{x}_{initial} = (400,\ -400, \underbrace{400, \ldots, 400}_{(d-2)\text{ times}})^T$ of the simplex and reflection coefficient $\alpha = 1$.



(a) The LCNM algorithm

(b) The LCNM+PC algorithm

Figure 5.2: Path of the minimum vertex $\mathbf{x}_1^q$ of the simplices for the quadratic objective function with global minimum inside of the region.

Figure 5.2 displays the trajectory of the minimum vertex $\mathbf{x}_1^q$ of each qth simplex for the case of two dimensions, when the original LCNM algorithm and the LCNM+PC method are applied to the problem. Figure 5.2(a) shows that the LCNM algorithm converges to the point $\mathbf{x}_{opt} = (-4.615, -3.077)^T$ on the boundary of the constraint l_1. However, this is not an optimum point of the constrained problem, because the point $(-4.615, -3.077)^T$ violates the Kuhn-Tucker (KT) necessary conditions.

Figure 5.2(b) also shows the results for LCNM+PC method showing that it converges to the minimum point $\mathbf{x}_{opt}^{s_1}$ on the boundary of the constraint l_2, whose coordinates correspond to $(-1.1, 4.25)^T$ during the first stage. In the second stage, the modified method converges to the global minimum point $\mathbf{x}_{opt} = (0, 0)^T$.

Table 5.2: Summary of Experiment 2.

	d	2	3	4	5	6	7	8
LCNM	NE	142	8537	1848	6445	630	3651	1585
	DTP	5.55	4.5E-162	5.55	2.48E-27	8.94	7.82E-15	8.94
LCNM	NE	3571	5841	819	3406	2824	3213	4304
+PC	DTP	1.2E-244	8.5E-163	7.67E-14	6.20E-24	2.29E-24	1.06E-14	7.32E-13
SM	NE	3340	∞	1848	∞	∞	∞	∞
	DTP	1.6E-162		5.547				

According to Table 5.2, the LCNM+PC method had a better performance than the LCNM method. From the results we found that the LCNM+PC method identified the global minimum point for each d-dimensional case, whereas the LCNM method converged to local minimum on the boundary of the feasible region in only the cases when the dimension of the problem is even. It can also be pointed out the SM did not converge in most cases.

5.3.3 Experiment 3

Powell singular function.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1+10x_2)^2+5(x_3-x_4)^2+(x_2-2x_3)^4+10(x_1-x_4)^4\right]$$

subject to:

$$\begin{aligned} l_1\!: &\quad x_1+x_2+x_3 &\geq b_1 \\ l_2\!: &\quad 2x_1+x_2+x_3+x_4 &\geq b_2 \end{aligned}$$

The Powell singular function has its unconstrained global minimum solution at the origin with a value function of $f(\mathbf{x}_{opt})=0$ (Moré et al. 1981). In this experiment, we set the initial point at $(100,100,100,100)^T$ for building the initial simplex, the stopping parameters were fixed as $\eta=0.1$, $\eta_1=\eta_2=10^{-6}$ and $\Delta=10^{-5}$ and the coefficient of reflection $\alpha=1$.

Two cases were considered: the first, when the constrained global minimum is on the boundary of the feasible region, whereby $b_1=3$ and $b_2=8$. The second case is when $b_1=-3$ and $b_2=-4$. In this latter case, the constrained global optimum corresponds to an interior point, whose coordinates are $(0,0,0,0)^T$. Observe that the SM satisfactorily works for the second case only.

Table 5.3: Summary of Experiment 3.

	b_1	b_2	NE	$f(\mathbf{x}_{opt})$	$x_{1,opt}$	$x_{2,opt}$	$x_{3,opt}$	$x_{4,opt}$
LCNM	3	8	626	17.4084	2.742	-0.194	0.592	2.119
LCNM+PC			785	17.4084	2.742	-0.194	0.592	2.119
SM			11899	140.88	3.773	0.112	-0.953	2.296
LCNM	-3	-4	557	2.5996	-1.373	0.110	-0.424	-0.940
LCNM+PC			2110	7.153E-58	-4.253E-15	4.253E-15	-2.192E-15	-2.192E-15
SM			3402	3.591E-63	-1.179E-16	1.793E-17	-4.271E-17	-4.271E-17

As can be seen in Table 5.3, both the LCNM method and the LCNM+PC method converge to the same local minimum when the global minimum is on the boundary of the feasible region. However, the LCNM+PC method requires more NE than the LCNM method.

In the second case, the LCNM+PC method obtained better performance than the LCNM method, because the LCNM+PC method converged to the constrained global optimum point, whilst the LCNM method converged to a local minimum on the boundary of the feasible region.

5.3.4 Experiment 4

Wood function.

$$\min_{\mathrm{x}\in\mathbb{R}^4} \left[100(x_2 - x_1^2)^2 + (1-x_1)^2 + 90(x_4 - x_3^2)^2 + (1-x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2\right]$$

subject to:

$$\begin{aligned} l_1&: \quad x_1 + x_2 + x_3 & \geq \; b_1 \\ l_2&: \quad 2x_1 + x_2 + x_3 + x_4 & \geq \; b_2 \end{aligned}$$

The unconstrained global minimum of this function is located at $(1,1,1,1)^T$ with function value zero (Moré et al. 1981). An initial point at $(100,100,100,100)^T$ was used for building the initial feasible simplex for the two first cases, whilst the last case was carried out with an initial point at $(10,10,10,10)^T$. The parameters of stopping rule were set as $\eta = 0.1$, $\eta_1 = \eta_2 = 10^{-6}$ and $\Delta = 10^{-5}$. With respect to the parameters b_1 and b_2, they were fixed at different values for the two cases. The first case, where $b_1 = 3$ and $b_2 = 5$, the constrained global minimum point is on both boundaries of the linear constraints, whereas for the other cases, the constrained global minima are inside the feasible region.

Table 5.4: Summary of Experiment 4.

	b_1	b_2	α	NE	$f(\mathbf{x}_{opt})$	$x_{1,opt}$	$x_{2,opt}$	$x_{3,opt}$	$x_{4,opt}$
LCNM	3	5	0.99	468	196.037	-1.497	2.578	1.919	3.497
LCNM+PC			0.99	612	3.618E-11	1.000	1.000	0.999	0.999
SM			0.99	4652	8.813	0.889	0.806	1.287	1.627
LCNM	-3	-5	1	811	2.502E-11	1.000	1.000	0.999	0.999
LCNM+PC			1	1022	1.355E-11	1.000	1.000	0.999	0.999
SM			1	810	3.939	-0.969	0.949	1.000	1.000
LCNM	2	4	0.99	317	59.624	0.691	1.779E-15	1.309	1.309
LCNM+PC			0.99	747	1.912E-11	0.999	0.999	1.000	0.999
SM			0.99	∞					
SM			0.9	548	2.545E-11	1.000	1.000	0.999	1.000

Table 5.4 gives a summary report of both cases. As is displayed in the table, the LCNM+PC had better performance than the other methods for all cases, because it reached the constrained global optimum satisfactorily.

5.4 Comparison of the methods

A comparison of the LCNM method, the LCNM+PC method and the SM is presented in this section through a set of test problems. For this study a Performance Measure (PM) was used that depends on the NE, the DTP and the feasibility of the obtained solution.

Let PM_i be the PM of each ith design point

$$PM_i = NE_i + w_1\ DTP_i + w_2\ (1 - F_i) \quad \forall i, \tag{5.3}$$

where NE is the number of function evaluations that are carried out in finding the optimum, DTP is the distance to the true point, F_i indicates if the found solution is feasible ($F_i = 1$) or infeasible ($F_i = 0$), and w_1=10 and w_2=4000 are weighted penalty factors. Here our interest is focused on minimizing PM.

It is worthwhile pointing out that some settings of the methods caused a large number of iterations. Hence, the experiments under these situations were artificially stopped when a set maximum NE in the experiment was reached and, these abrupt stoppings are indicated by the symbol "+" beside the NE.

These performance measures were studied as a function of the reflection coefficient α, the step size parameter τ and the different methods. For this particular study, a complete factorial experiment was used where each factor was defined at three levels (Myers and Montgomery 2002), (Khuri and Cornell 1996).

The coded factors x_α, x_τ and x_m of our experimental design are given by

$$x_\alpha = \begin{cases} -1 & \alpha = 0.90 \\ 0 & \alpha = 0.95 \\ 1 & \alpha = 1.00 \end{cases} ; \quad x_\tau = \begin{cases} -1 & \tau = 0.5 \\ 0 & \tau = 1.0 \\ 1 & \tau = 1.5 \end{cases} \text{ and } x_m = \begin{cases} -1 & \text{SM} \\ 0 & \text{LCNM} \\ 1 & \text{LCNM+PC} \end{cases}$$

Table 5.5 displays the set of design experiment points. The reflection coefficient α, step size parameter τ, the algorithm of optimization and the coded factors x_α, x_τ and x_m are represented in the table by row.

Table 5.5: Table of setting for the experimental design.

Exp	α	τ	Method	x_α	x_τ	x_m
1	0.90	0.5	SM	-1	-1	-1
2	0.95	0.5	SM	0	-1	-1
3	1.00	0.5	SM	1	-1	-1
4	0.90	1.0	SM	-1	0	-1
5	0.95	1.0	SM	0	0	-1
6	1.00	1.0	SM	1	0	-1
7	0.90	1.5	SM	-1	1	-1
8	0.95	1.5	SM	0	1	-1
9	1.00	1.5	SM	1	1	-1
10	0.90	0.5	LCNM	-1	-1	0
11	0.95	0.5	LCNM	0	-1	0
12	1.00	0.5	LCNM	1	-1	0
13	0.90	1.0	LCNM	-1	0	0
14	0.95	1.0	LCNM	0	0	0
15	1.00	1.0	LCNM	1	0	0
16	0.90	1.5	LCNM	-1	1	0
17	0.95	1.5	LCNM	0	1	0
18	1.00	1.5	LCNM	1	1	0
19	0.90	0.5	LCNM+PC	-1	-1	1
20	0.95	0.5	LCNM+PC	0	-1	1
21	1.00	0.5	LCNM+PC	1	-1	1
22	0.90	1.0	LCNM+PC	-1	0	1
23	0.95	1.0	LCNM+PC	0	0	1
24	1.00	1.0	LCNM+PC	1	0	1
25	0.90	1.5	LCNM+PC	-1	1	1
26	0.95	1.5	LCNM+PC	0	1	1
27	1.00	1.5	LCNM+PC	1	1	1

5.4.1 Description of test problems

The test problems were divided into two main categories conforming with the solid angle feasible cone that can be calculated through the scalar product of the normal vector of each hyperplane of boundary constraint, these are: obtuse solid angle feasible cone and non-obtuse solid angle feasible cone.

Because there exist several local minima in some of the following test problems, the true point (TP) of each test was taken to be the best local minimum or the also called constrained global minimum, whose coordinates are indicated in the description of the test problems.

Figure 5.3 illustrates a taxonomical structure of the test problems that were carried out for contrasting the SM, the LCNM method and the LCNM+PC method. These test problems were categorized by the feasible region, the location of the best local minimum and the type of function.

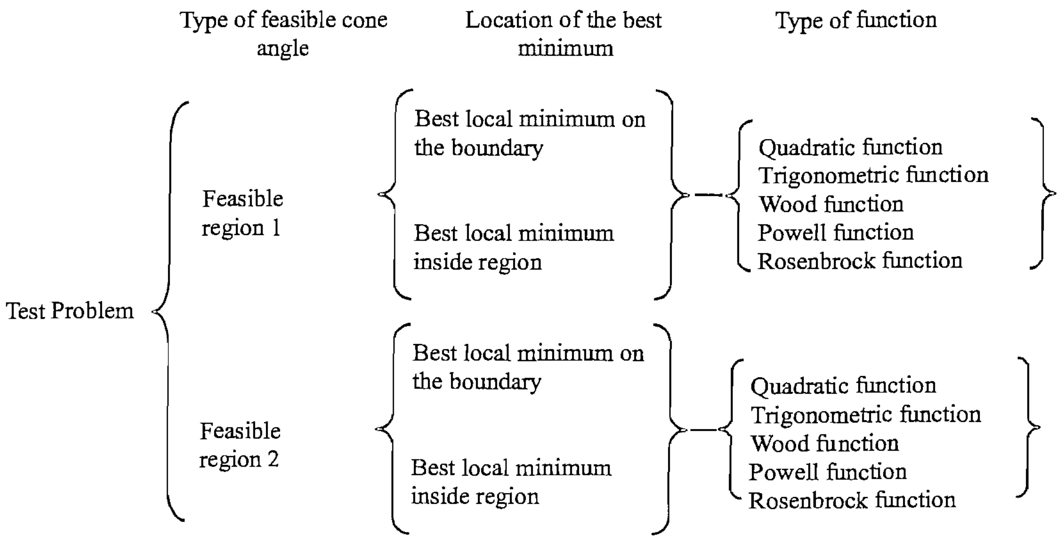


Figure 5.3: Taxonomic of the test problems when the objective function is deterministic.

The feasible region of each test problem was selected by locating the constrained global minimum of the objective function either on its boundary or inside it. For the objective functions, we selected simple functions, such as strictly convex functions and complicated functions, such as Rosenbrock, Wood and Powell functions, which may possess several local minima on the boundary, depending on the feasible region.

Figure 5.4 shows both the contour line plot and the three dimensional plot of a type of

Rosenbrock function, whose mathematical expression is given by $f(x_1, x_2) = (x_2 - x_1^2)^2 + (1 - x_2)^2$ for all x_1, x_2. As can be seen from the figure, this function possesses a global minimum at the point $(1,1)^T$. However, a constrained optimization problem based on this class of function could present several local minima. This fact was taken into account in formulating test problems.

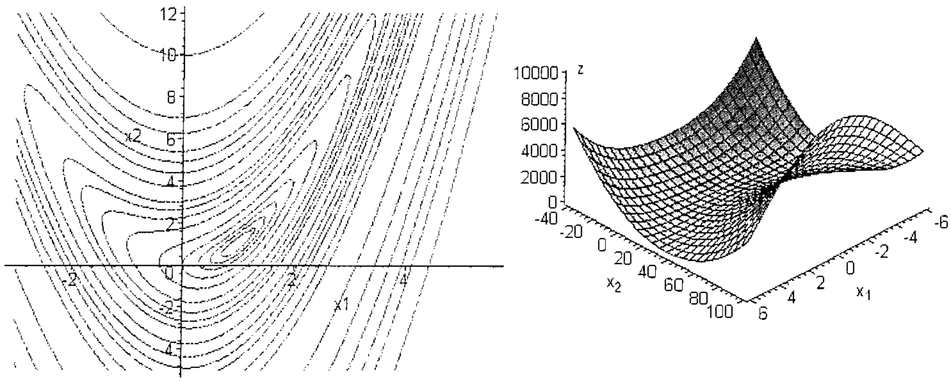


Figure 5.4: Rosenbrock function.

Figure 5.5 depicts the contour line plot and three dimensional plot of the employed trigonometric function for two independent variables, whose mathematical expression is shown in Equation (5.4)

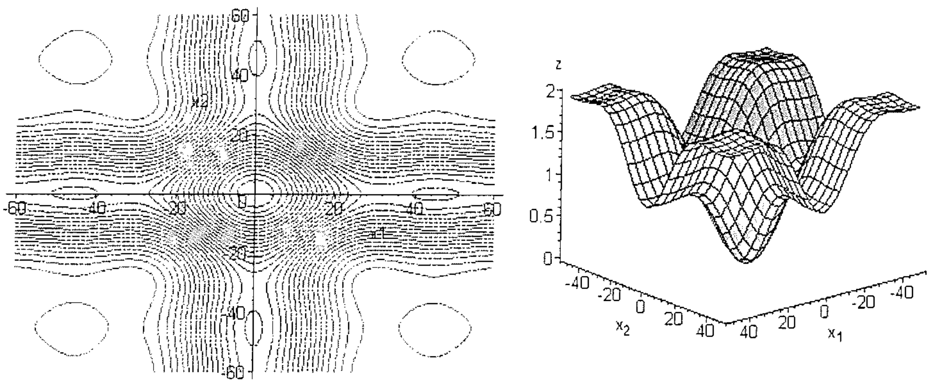


Figure 5.5: Trigonometric function.

$$f(x_1, x_2) = \begin{cases} 2 - \sum_{i=1}^{2} \left[\frac{\sin(0.1x_i)}{0.1x_i}\right]^2 & \text{if } (x_1, x_2) \neq (0,0), \\ 0 & \text{if } (x_1, x_2) = (0,0) \end{cases} \tag{5.4}$$

This function has multiple local minima as can be identified in the picture. Note that its global minimum is at the origin.

Figure 5.6 illustrates a set of three dimensional plots of the function of Wood where two of the four variables were fixed at one. Its global minimum is located at the point $(1,1,1,1)^T$. The mathematical expression of the Wood function is given by Equation (5.5).

$$f(\mathbf{x}) = 100(x_2-x_1^2)^2+(1-x_1)^2+90(x_4-x_3^2)^2+(1-x_3)^2+10(x_2+x_4-2)^2+10(x_2-x_4)^2 \quad \forall \mathbf{x} \tag{5.5}$$

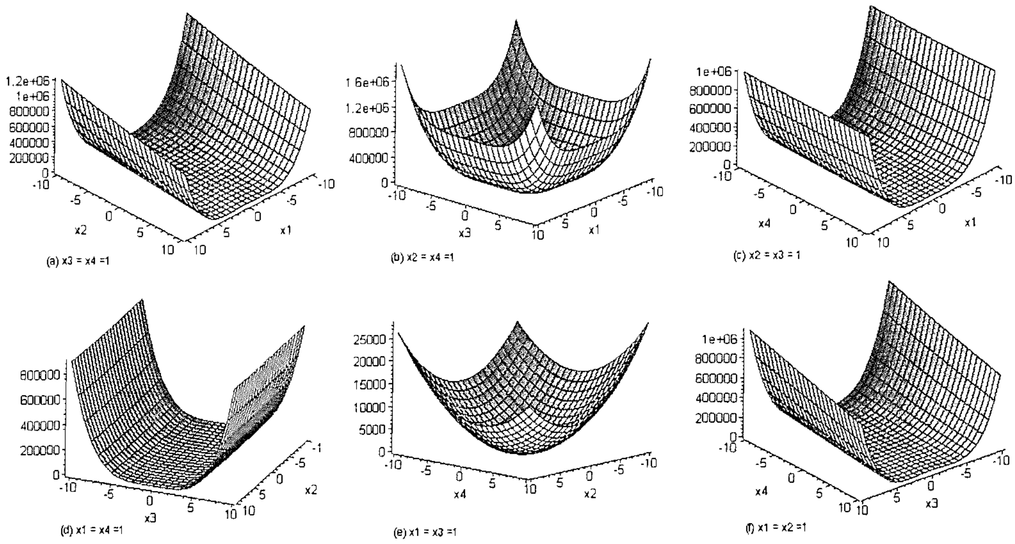


Figure 5.6: Three dimension plots of the Wood function.

As can be seen from Figure 5.6, the function is flat close to its minimum, when two variables are fixed at one. This fact could cause a slow convergence of the optimization algorithm, when the feasible region includes this flat region.

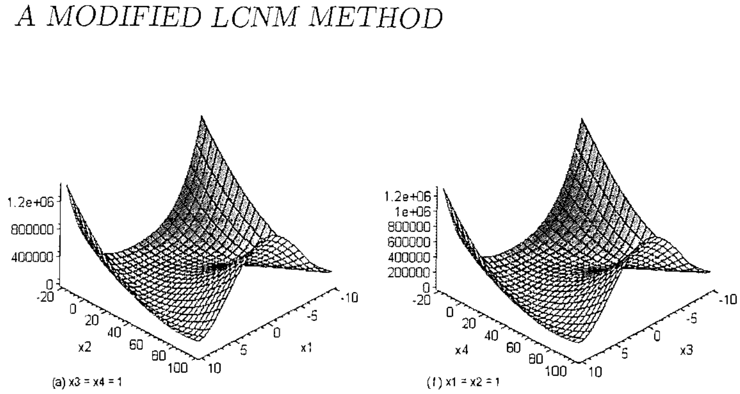


Figure 5.7: Wood function for the case a) $x_3 = x_4 = 1$ and f) $x_1 = x_2 = 1$.

The function of Wood contains terms similar to the Rosenbrock function. It behaves as the function of Rosenbrock, when x_1 and x_2 (x_3 and x_4) are allowed vary at fixed values of x_3 and x_4 (x_1 and x_2). This feature follows easily from Equation (5.5) and it is graphically represented in Figure 5.7, in which are displayed the cases when $x_1 = x_2 = 1$ and $x_3 = x_4 = 1$.

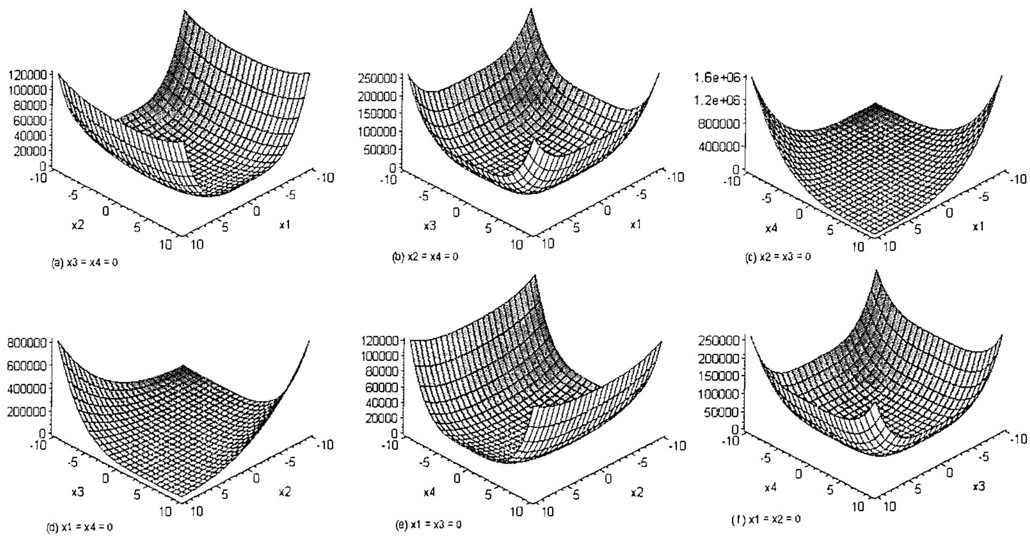


Figure 5.8: Three dimension plots of the Powell function.

A similar feature to the function of Wood can be appreciated in the function of Powell, when it is in the neighbourhood of its global optimum. It is depicted in Figure 5.8. Observe that the function of Powell flats near to its global optimum $(0,0,0,0)^T$ and its mathematical expression is given by Equation (5.6). Nevertheless, Powell function does not contain Rosenbrock function terms.

$$f(\mathbf{x}) = (x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4 \quad \forall \mathbf{x} \tag{5.6}$$

Case study 1: obtuse solid angle feasible cone

The feasible region $\mathcal{F}_1$ for this case is given by,

$$\begin{aligned} C_1: &\quad \sum_{i=1}^{d} x_i \geq b_1 \\ C_2: &\quad 2x_1 + \sum_{i=2}^{d} x_i \geq b_2 \end{aligned}$$

where b_1 and b_2 were selected so as to give at least a local minimum on the boundary or inside the feasible region.

For all test problems of this case study, the initial point of the simplex was taken to be the point $10 \cdot \mathbf{1}_d$, where $\mathbf{1}_d$ is the unit vector of dimension d.

Test Problem 1

Quadratic objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = 3 \wedge b_2 = 5\}$

The unconstrained global minimum is located at $\mathbf{x}_{opt} = (0, 0, \ldots, 0)^T$ and $f(\mathbf{x}_{opt}) = 0$. Constrained global minimum occur at the following points. These are verified in Appendix A on page 211.

$$\mathbf{x}_{local}^T = \begin{cases} (2, 1) & d = 2, \\ (1.42857,\ 0.714286,\ 0.714286,\ 0.714286) & d = 4, \\ (1.11111,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555) & d = 6. \end{cases}$$

Test Problem 2

Quadratic objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = -3 \wedge b_2 = -5\}$.

The constrained global minimum is located at $\mathbf{x}_{local} = (0, 0, \ldots, 0)^T$.

Test Problem 3

Trigonometric objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[d - \sum_{i=1}^{d} g_i^2 \right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = 3 \wedge b_2 = 5\}$,

where g_i is given by

$$g_i = \begin{cases} \frac{\sin(0.1x_i)}{0.1x_i} & \text{if } x_i \neq 0 \\ 1 & \text{otherwise} \end{cases} \tag{5.7}$$

The unconstrained global minimum is located at $\mathbf{x}_{opt} = (0, 0, \ldots, 0)^T$ and $f(\mathbf{x}_{opt}) = 0$. In this group of tests, the constrained global minimum points occur at

$$\mathbf{x}_{local}^T = \begin{cases} (2, 1) & d = 2, \\ (1.42857,\ 0.714286,\ 0.714286,\ 0.714286) & d = 4, \\ (1.1111,\ 0.5555,\ 0.5555,\ 0.5555,\ 0.5555,\ 0.5555,\ 0.5555) & d = 6. \end{cases}$$

Test Problem 4

Trigonometric objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[d - \sum_{i=1}^{d} g_i^2 \right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = -3 \wedge b_2 = -5\}$, where g_i is given by Equation (5.7), In this case, the constrained global minimum is located at $\mathbf{x}_{local} = (0, 0, \ldots, 0)^T$.

Test Problem 5

Extended Rosenbrock objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x} \in \mathbb{R}^d} \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = 3 \wedge b_2 = 5 \;\; \forall d = 2, 4 \text{ and } b_1 = 6 \wedge b_2 = 7 \;\; \forall d = 6\}$, where d is an even integer number.

The unconstrained global minimum is located at $\mathbf{x}_{opt} = (1, 1, \ldots, 1)^T$ and $f(\mathbf{x}_{opt}) = 0$.

The constrained global minima for each test are

$$\mathbf{x}_{local}^T = \begin{cases} (-3.447634, 11.895268) & d = 2, \\ (1, 1, 1, 1) & d = 4, \\ (1, 1, 1, 1, 1, 1) & d = 6. \end{cases},$$

which is verified for $d = 2$ in Appendix A on page 212.

Test Problem 6

Extended Rosenbrock objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x} \in \mathbb{R}^d} \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = -3 \wedge b_2 = -5\}$, where d is an even integer number.

The constrained global minimum for this group of tests corresponds to the point $\mathbf{x}_{local} = (1, 1, \ldots, 1)^T$.

Test Problem 7

Wood objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = 3 \wedge b_2 = 5\}$.

The unconstrained global minimum is located at $\mathbf{x}_{opt} = (1, 1, 1, 1)^T$ and $f(\mathbf{x}_{opt}) = 0$, and its constrained global minimum is located at the same coordinates.

Test Problem 8

Wood objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = -3 \wedge b_2 = -5\}$.

In this test problem the constrained global minimum was $\mathbf{x}_{local} = (1, 1, 1, 1)^T$.

Test Problem 9

Powell singular objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_1 \text{ with } b_1 = 3 \wedge b_2 = 5\}$.

The unconstrained global minimum is located at $\mathbf{x}_{opt} = (0, 0, 0, 0)^T$ and $f(\mathbf{x}_{opt}) = 0$.

The constrained global minimum is given by
$\mathbf{x}_{local} = (1.715358, -0.132167, 0.476726, 1.224726)^T$, which is verified in Appendix A on page 213.

Test Problem 10

Powell singular objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1+10x_2)^2+5(x_3-x_4)^2+(x_2-2x_3)^4+10(x_1-x_4)^4\right]$$

subject to $\{\mathbf{x}\in\mathbb{R}^4 \mid \mathbf{x}\in\mathcal{F}_1 \text{ with } b_1=-3 \wedge b_2=-5\}$.

The constrained global minimum is $\mathbf{x}_{local}=(0,0,0,0)^T$.

Case study 2: non-obtuse solid angle feasible cone

The feasible region $\mathcal{F}_2$ for this set of test problems is given by,

$$\begin{aligned} C_3:&\quad 2x_1-x_2+\sum_{i=3}^{d} x_i \geq b_3 \\ C_4:&\quad -x_1+2x_2+\sum_{i=3}^{d} x_i \geq b_4 \end{aligned}$$

where b_3 and b_4 were selected to give at least a local minimum on the boundary of the feasible region or inside the feasible region.

As initial point for all test problems the point $20\cdot\mathbf{1}_d$ was used.

Test Problem 11

Quadratic objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $\{\mathbf{x}\in\mathbb{R}^d \mid \mathbf{x}\in\mathcal{F}_2 \text{ with } b_3=2 \wedge b_4=2\}$

The constrained global minimum occur at the following points

$$\mathbf{x}_{local}^T=\begin{cases} (2,2) & d=2, \\ (0.4,\ 0.4,\ 0.8,\ 0.8) & d=4, \\ (0.22222,\ 0.22222,\ 0.44444,\ 0.44444,\ 0.44444,\ 0.44444,\ 0.44444) & d=6. \end{cases}$$

Test Problem 12

Quadratic objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = -2 \wedge b_4 = -2\}$.

The constrained global minimum is located at $\mathbf{x}_{local} = (0, 0, \ldots, 0)^T$.

Test Problem 13

Trigonometric objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[d - \sum_{i=1}^{d} g_i^2 \right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = 2 \wedge b_4 = 2\}$, where g_i is given by Equation (5.7).

The constrained global minimum points are

$$\mathbf{x}_{local}^T = \begin{cases} (2, 2) & d = 2, \\ (0.4,\ 0.4,\ 0.8,\ 0.8) & d = 4, \\ (0.22222,\ 0.22222,\ 0.44444,\ 0.44444,\ 0.44444,\ 0.44444,\ 0.44444) & d = 6. \end{cases}$$

Test Problem 14

Trigonometric objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left[d - \sum_{i=1}^{d} g_i^2 \right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = -2 \wedge b_4 = -2\}$, where g_i is given by Equation (5.7).

In this case, the constrained global minimum is located at $\mathbf{x}_{local} = (0, 0, \ldots, 0)^T$.

Test Problem 15

Extended Rosenbrock objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = d - 1 \wedge b_4 = d - 1\}$, where d is an even integer number.

The constrained global minimum for each test is

$$\mathbf{x}_{local}^T = \begin{cases} (1,1) & d = 2, \\ (1,1,1,1) & d = 4, \\ (1,1,1,1,1,1) & d = 6. \end{cases}$$

Test Problem 16

Extended Rosenbrock objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d/2} \left[100(x_{2i} - x_{2i-1}^2)^2 + (1 - x_{2i-1})^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^d \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = -2 \wedge b_4 = -2\}$, where d is an even integer number.

The constrained global minimum for this group of tests corresponds to the point $\mathbf{x}_{local} = (1, 1, \ldots, 1)^T$.

Test Problem 17

Wood objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = 3 \wedge b_4 = 3\}$.

The constrained global minimum is located at $\mathbf{x}_{local} = (1, 1, 1, 1)^T$.

Test Problem 18

Wood objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[100(x_2 - x_1^2)^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10(x_2 + x_4 - 2)^2 + 10(x_2 - x_4)^2\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = -2 \wedge b_4 = -2\}$.

The constrained global minimum is $\mathbf{x}_{local} = (1, 1, 1, 1)^T$.

Test Problem 19

Powell singular objective function with its best optimum point on the boundary.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = 3 \wedge b_4 = 3\}$.

The constrained global minimum is given by $\mathbf{x}_{local} = (1, 1, 1, 1)^T$.

Test Problem 20

Powell singular objective function with its best optimum point inside the feasible region.

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4\right]$$

subject to $\{\mathbf{x} \in \mathbb{R}^4 \mid \mathbf{x} \in \mathcal{F}_2 \text{ with } b_3 = -2 \wedge b_4 = -2\}$.

The constrained global minimum is $\mathbf{x}_{local} = (0, 0, \ldots, 0)^T$.

5.4.2 Plots of experiments

We give a set of plots of the performance measure for each test problem defined in the previous subsection. The SM converged to infeasible points in 21 of the 44 test problems carried out and reported in Appendix D. In contrast, both the LCNM method and the LCNM+PC method always converged to a feasible point. This fact was taken into account in the function of PM through the factor w_2=4000 (see Equation (5.3)).

To facilitate the reading of the test problem plots and their summary reports, we use the notation TP#(d), where # is the number of the test problem and d indicates its dimension. For instance, TP1(4) means the Test Problem 1 with dimension 4, TP2(6) represents the Test Problem 2 with dimension 6 and so forth.

Furthermore, the vertical axis of each plot represents the PM value, whilst the horizontal axis corresponds to the 27 design points grouped by method, that is, the first group of 9-design points corresponds to the set of PM yielded by the SM, the second group, from 10 to 18, is the set of PM obtained by the LCNM method, and the last group corresponds to the experiments performed on the LCNM+PC method.

Case study 1: obtuse solid angle feasible cone

Before describing the graphs, we should briefly comment that the PM of the SM were higher than those obtained by the LCNM and the LCNM+PC method in the majority of the test problems. When the objective function is quadratic, the LCNM method gave a better performance than the LCNM+PC method. However, for the rest of the test problems, we did not observe a significant advantage of the LCNM+PC method in comparison with the LCNM method. This may be verified in the set of tables presented in Subsection 5.4.5.

According to Figure 5.9 and Figure 5.10, the LCNM method and the LCNM+PC method had a higher performance when the global minimum is inside the feasible region than in the case when the global minimum is on the boundary. In addition, there is no significant difference between both methods in the indicated test problems. Results of these experiments are shown in Table 5.8 on page 152.

Furthermore, the LCNM and the LCNM+PC method show a high sensitivity to changing of the design point in the TP2(4) and TP2(6).

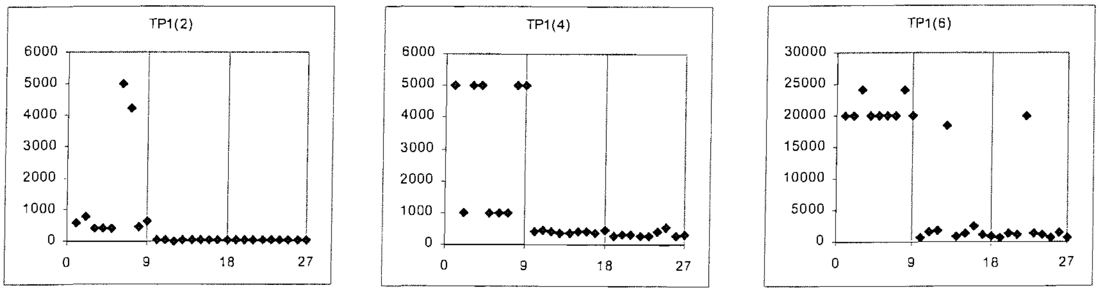


Figure 5.9: Performance of the methods on the test problem TP1.

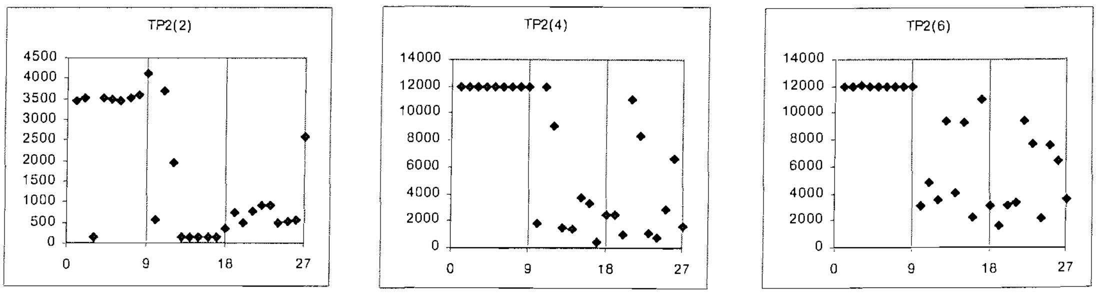


Figure 5.10: Performance of the methods on the test problem TP2.

From Figure 5.11 and Figure 5.12, it may be seen that the LCNM method and the LCNM+PC method had approximately the same PM for each test problem. However, both methods had a higher PM when the global minimum is inside the feasible region. A summary report of these tests is given in Table 5.9 on page 153.

Another aspect that can be appreciated from the figures, is the fact that the LCNM and the LCNM+PC method generally failed in the TP4(2), TP4(4) and TP4(6) cases. Nonetheless, for some design points in these test problems the methods adequately converged.

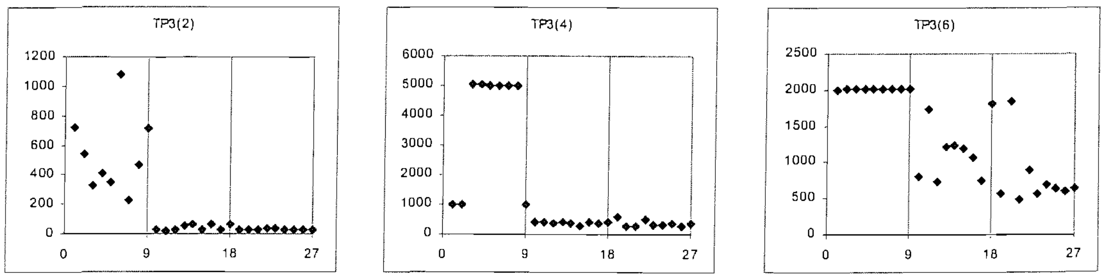


Figure 5.11: Performance of the methods on the test problem TP3.

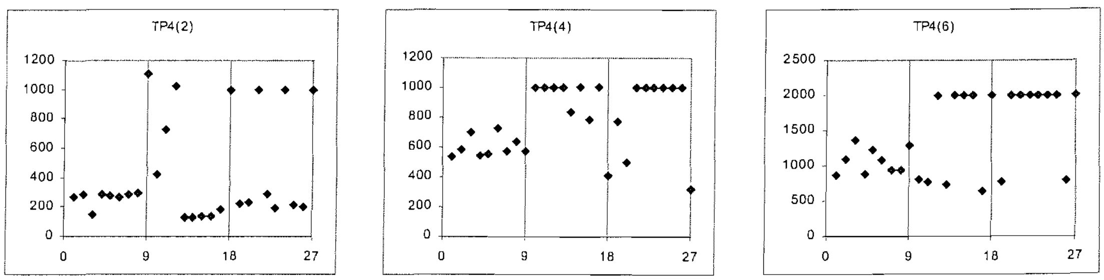


Figure 5.12: Performance of the methods on the test problem TP4.

As are shown in Figure 5.13 and Figure 5.14, in each test problem the PM of the LCNM method is as high as that of the LCNM+PC method. In addition, the largest PM of both methods occurred in the cases TP5(4) and TP5(6) whose constrained global minimum are on the boundary of the feasible region. However, for most of the design points, the LCNM and its modified version have the same behaviour.

We must remark that the SM evidenced better performance than our methods for some design points and this fact can be clearly appreciated in Table 5.10 (on page 154).

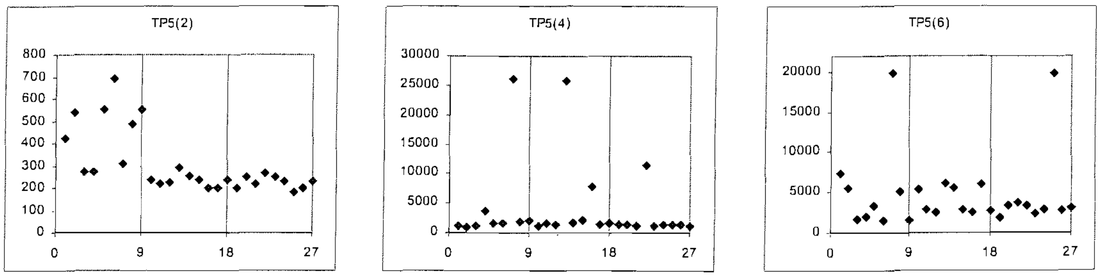


Figure 5.13: Performance of the methods on the test problem TP5.

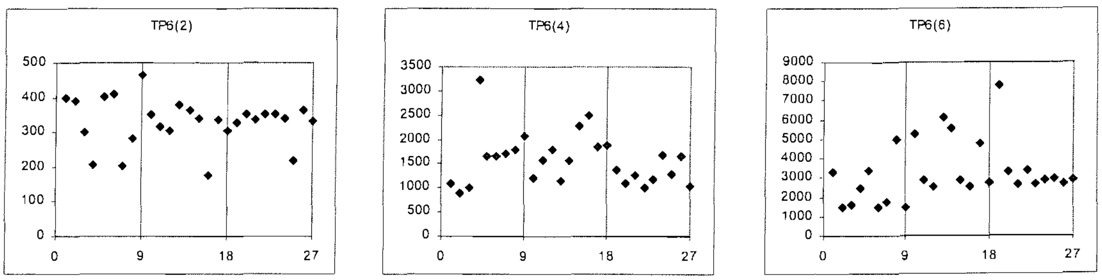


Figure 5.14: Performance of the methods on the test problem TP6.

From Figure 5.15 and Figure 5.16, it may be concluded that the PM of the LCNM method and the PM of the LCNM+PC method are approximately equal for each test problem. However, both methods had the highest PM when the global minimum is inside the feasible region. A summary report of these tests is given in Table 5.11 on page 155.

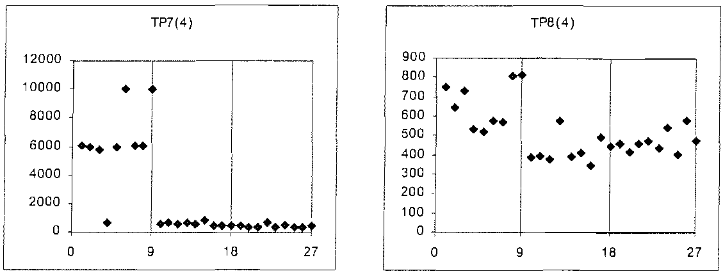


Figure 5.15: Performance of the methods on the test problems TP7(4) and TP8(4).

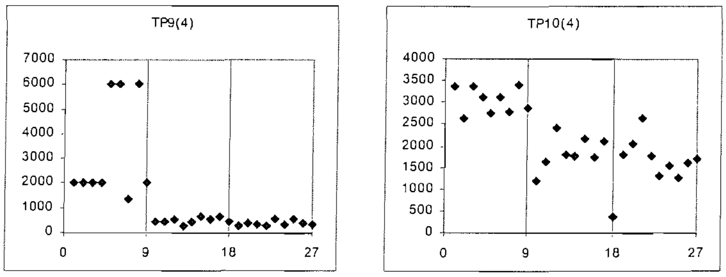


Figure 5.16: Performance of the methods on the test problems TP9(4) and TP10(4).

Case study 2: non-obtuse solid angle feasible cone

As may be seen, in this group of experiments, we can note an advantage of the LCNM method and the LCNM+PC method with respect to the SM from the reported PM's. Nevertheless, as in the previous case, our methods have the same behaviour, in terms of average, even in the cases of dimension two, where their PM are approximately equal (see numerical report in Subsection 5.4.5 and for more details in Appendix D).

According to Figure 5.17 and Figure 5.18, the PM of the LCNM method and the PM of the LCNM+PC method are approximately equal for each test problem. However, the figures show that both methods had better performance when the global minimum is on the boundary of the feasible region than when the global optimum is inside the feasible region. Numerical results are reported in Table 5.12 on page 156.

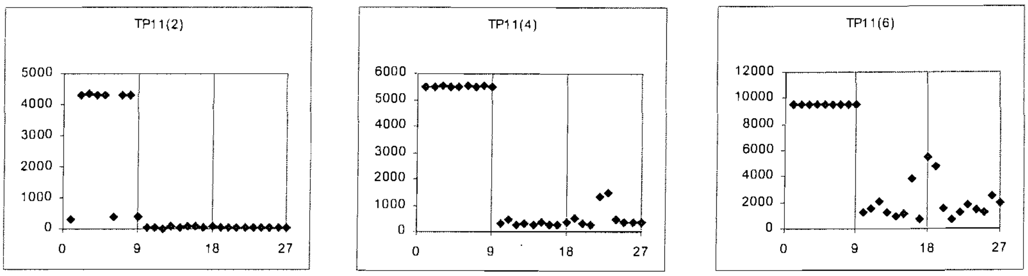


Figure 5.17: Performance of the methods on the test problem TP11.

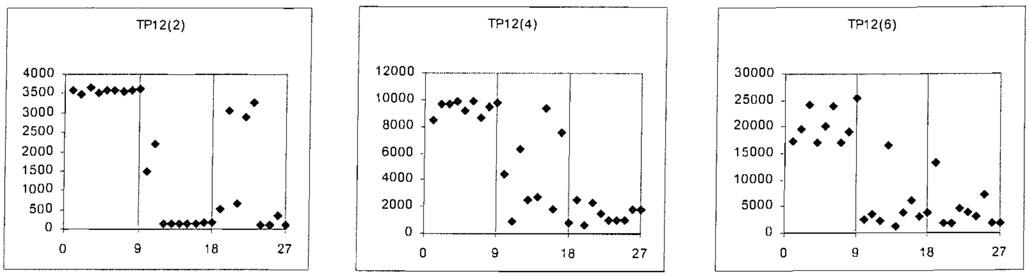


Figure 5.18: Performance of the methods on the test problem TP12.

As can be seen in Figure 5.19 and Figure 5.20, the LCNM method and the LCNM+PC method had approximately the same PM for each test problem. However, both methods had higher PM when the global minimum is inside the feasible region than in the cases with global optimum on the boundary of the feasible region. A summary report of these tests is given in Table 5.13 on page 157.

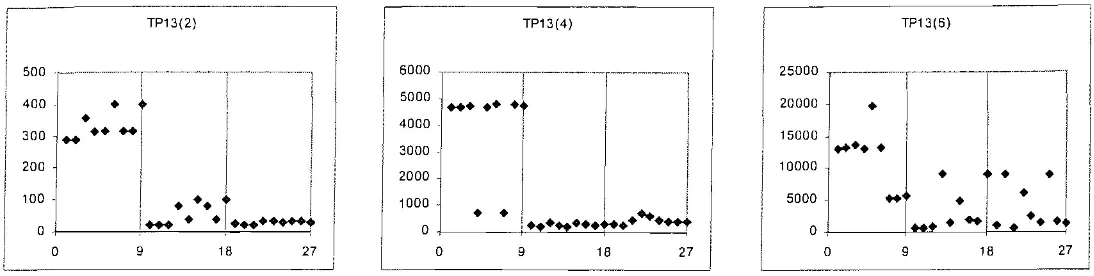


Figure 5.19: Performance of the methods on the test problem TP13.

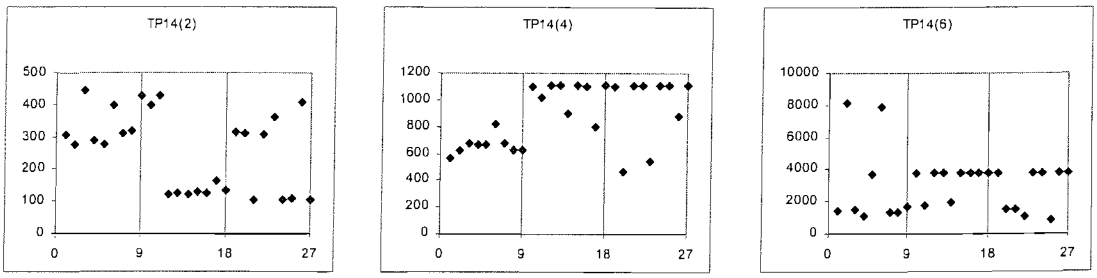
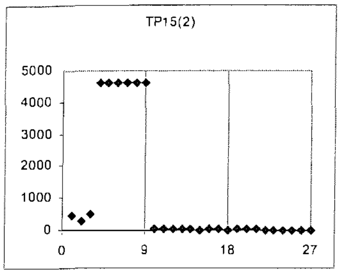
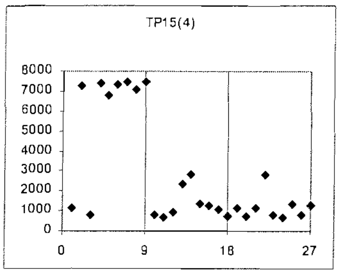


Figure 5.20: Performance of the methods on the test problem TP14.

It may be seen from Figure 5.21 and Figure 5.22 that the PM of the LCNM method is as high as the PM of the LCNM+PC method for each test, and they presented better performance when the global minimum point is inside the feasible region than when the minimum is on the boundary. A report of these tests is displayed in Table 5.14 on page 158.





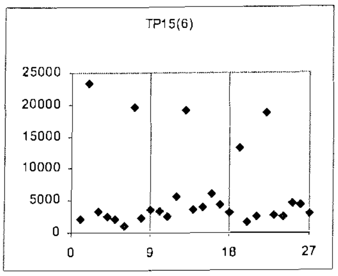
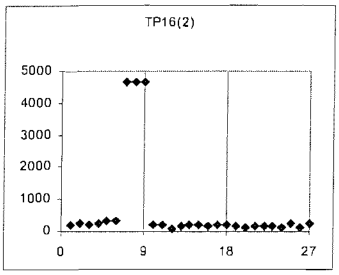
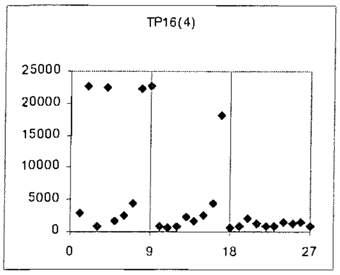


Figure 5.21: Performance of the methods on the test problem TP15.





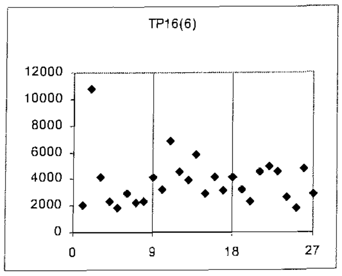


Figure 5.22: Performance of the methods on the test problem TP16.

As are shown in Figure 5.23 and Figure 5.24, in each test problem the PM of the LCNM method is as high as that of the LCNM+PC method, in terms of average. Nevertheless, the PM of both methods was larger in the case when the global minimum is inside the feasible region than when the best minimum is on the boundary of the feasible region. Table 5.15 (on page 159) shows a summary of the these results.

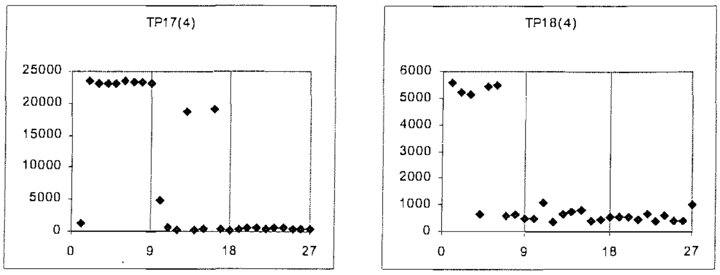


Figure 5.23: Performance of the methods on the test problems TP17(4) and TP18(4).

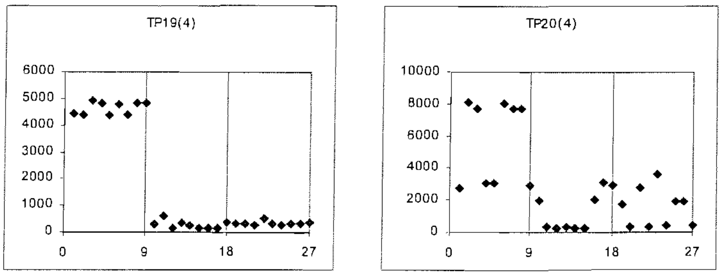


Figure 5.24: Performance of the methods on the test problems TP19(4) and TP20(4).

5.4.3 Average of performance measure by experimental design point

Here we display the average PM figures by solid angle, dimension and experimental design point. This latter is completely specified in Table 5.5 on page 126.

Table 5.6: Average of PM for obtuse and non-obtuse solid angle

	o∡			no∡		
Exp	*d*=2	*d*=4	*d*=6	*d*=2	*d*=4	*d*=6
1	976.5	3288.2	7586.9	853.0	3454.6	7571.5
2	1006.0	2767.8	7014.2	1488.0	9172.9	14089.3
3	267.0	3669.0	7111.9	1587.3	6313.9	9370.8
4	847.8	3582.1	6569.5	2215.8	7834.0	7597.0
5	911.0	3719.2	7010.8	2235.5	6461.7	9489.0
6	1818.7	4178.8	6346.9	1627.1	7262.8	9743.4
7	1464.4	5706.1	9431.4	2961.4	6350.2	9185.5
8	934.2	4245.2	8150.9	2965.9	8629.5	6653.9
9	1265.7	3846.7	6376.0	2354.8	8206.7	8333.7
10	270.9	873.4	2660.6	360.5	1531.1	2455.9
11	836.4	2023.2	2450.0	484.4	673.8	2802.4
12	593.6	1792.1	2178.3	►67.0	1099.9	3200.8
13	173.0	3353.8	7019.9	107.9	2872.7	8900.9
14	165.8	924.6	3202.8	95.6	1003.2	2538.0
15	149.7	1395.1	3265.1	112.3	1668.2	3423.3
16	►120.4	1814.8	2118.4	109.0	3088.9	4346.7
17	158.5	912.5	4046.9	108.1	3200.9	2806.6
18	336.0	886.0	2199.3	119.3	797.8	4900.6
19	255.6	992.0	2221.3	178.7	965.1	6555.6
20	224.4	771.9	2477.3	595.5	►632.8	2989.5
21	394.4	1895.5	2237.3	167.5	1062.9	►1944.6
22	312.6	2568.1	6520.2	576.0	1027.7	6101.5
23	298.4	►764.6	2750.8	647.8	1006.1	3198.5
24	353.3	841.3	►1955.6	72.8	682.4	2469.2
25	200.7	1009.8	5610.4	93.0	848.2	4129.5
26	231.3	1413.6	2414.5	158.6	880.9	3161.2
27	697.9	793.7	2127.6	86.5	786.2	2429.9

In Table 5.6 we denote the obtuse solid angle by o∡ and the non-obtuse by no∡. Additionally, the best average PM values are indicated by the symbol "►" for each dimension. As can be clearly seen from above table, the LCNM method and its premature version have better performance than the SM.

Comparing the LCNM method and the LCNM+PC method, the LCNM method has better average performance than its premature mode in two dimensional problems. However, the average PM of LCNM+PC method is lower than the average PM of the LCNM method.

Furthermore, the best average PM were obtained when the reflection coefficient α was fixed to 0.95 and 1.00 in dimension 4 and 6, respectively.

With respect to the step size τ, from Equation (3.38) on page 44, we have

$$\nu = \begin{cases} \max\left\{\tau\left|x_{i,1}\right| : i = 1, 2, \ldots d\right\} & if\ \mathbf{x}_1 \neq \mathbf{0}_d, \\ 1 & otherwise \end{cases}.$$

Because the initial point for the case study 1 was fixed to $10 \cdot \mathbf{1}_d$ and for the case study 2 was fixed to $20 \cdot \mathbf{1}_d$, it can be concluded that the best value of ν was 10 for the test problems of dimension 4 and 6, whose τ value corresponds to τ=1 when the solid angle is obtuse and τ=0.5 for non-obtuse solid angle. Nonetheless, for the test problems of dimension 2, the best setting of τ was 1.5 and 0.5, for the cases obtuse solid angle and non-obtuse solid angle, respectively.

As a result of the average PM reported in Table 5.6, we have that the values of 0.95 or 1 are adequate for setting the reflection coefficient α, and 1 for the step size τ.

5.4.4 Maximum performance measure by experimental design point

Another way of comparing the PM's is the analysis of the maximum values of the PM's. Table 5.7 depicts the maximum of PM's for both cases, obtuse and non-obtuse solid angle, by dimension and experimental design point.

Table 5.7: Maximum PM for obtuse and non-obtuse solid angle

	o∡			no∡		
Exp	d=2	d=4	d=6	d=2	d=4	d=6
1	3459.0	12000.0	20006.3	3573.0	8514.0	17255.0
2	3515.0	12000.0	20006.7	4290.0	23540.0	23295.6
3	406.0	12000.0	24027.4	4357.0	23081.2	24125.0
4	3525.0	12000.0	20007.6	4617.8	23013.6	17123.0
5	3492.0	12000.0	20008.9	4622.8	23137.9	20036.0
6	5000.0	12000.0	20008.3	4622.8	23466.7	23817.0
7	4233.1	26000.0	20009.2	4649.8	23360.0	19629.1
8	3608.0	12000.0	24019.0	4649.8	23398.1	19107.0
9	4111.8	12000.0	20016.4	4649.8	23038.5	25519.0
10	566.0	1860.0	5298.2	1479.0	4860.9	►3800.0
11	3710.0	12000.0	4799.0	2190.0	►1098.8	6844.3
12	1963.0	9093.0	3494.0	►123.9	6334.0	5728.6
13	378.0	25742.9	18484.4	185.9	18650.9	19025.4
14	363.0	1756.0	5547.5	193.9	2802.7	5834.5
15	338.0	3773.0	9255.0	195.9	9373.0	4892.0
16	►198.0	7723.5	►2505.2	186.9	19020.0	6184.0
17	336.0	2095.0	10971.0	193.9	17989.7	4454.2
18	1000.0	2413.0	3092.0	193.9	2969.0	9109.4
19	726.0	2447.0	7777.0	518.0	2535.0	13275.4
20	471.0	2050.0	3294.9	3077.0	2158.0	9100.0
21	1000.0	11031.0	3736.5	658.0	2800.0	4510.4
22	905.0	11388.6	20015.7	2887.0	2815.8	18740.2
23	924.0	►1316.0	7651.0	3260.0	3632.0	4529.0
24	1000.0	1667.0	2912.3	145.0	1383.0	3806.7
25	528.0	2935.0	19878.9	256.0	1924.0	9100.5
26	548.0	6583.0	6452.0	408.9	1944.0	4756.5
27	2575.0	1718.0	3532.0	240.0	1798.0	3800.5
	►TP5(2)	►TP10(4)	►TP6(6)	►TP14(2)	►TP18(4)	►TP14(6)

In Table 5.7, the minimum of the maximum PM values is emphasized by the symbol "►" for each dimension, and the corresponding test problem (by dimension) is also shown at the end of each column, details of which were given previously.

Note that all the minimum of the maximum PM values were yielded by the LCNM method, except in the case when the solid angle is obtuse and the dimension of the test problem is 4, whose minimum value was obtained by the LCNM+PC method.

The LCNM method adequately worked with a step size parameter setting of 0.5 for the test problems with non-obtuse solid angle, whilst, the LCNM method had its best performance when the step size was fixed to 1.5 for the test problems with obtuse solid angle.

5.4.5 Summary of comparison results

In this subsection we report the numerical results of the PM's for the case studies plotted in Subsection 5.4.2. Details of the NE's, the DTP's and the feasibility of the obtained solutions for the test problems carried out are shown in Appendix D.

Case study 1: obtuse solid angle feasible cone

From Table 5.8 to Table 5.11 are displayed the PM of the set of test problems carried out for the case studies with obtuse solid angle cone of the feasible region . Note that some figures in the tables are reported with a symbol "+", which means that the method was artificiality stopped at the indicated number of function evaluations.

Table 5.8: Summary result of the performance of the families of test problems TP1 and TP2.

Exp	TP1(2)	TP1(4)	TP1(6)	TP2(2)	TP2(4)	TP2(6)
1	582	5006+	20006.3+	3459	12000+	12000+
2	770	1002.2+	20006.7+	3515	12000+	12000+
3	406	5017.3+	24027.4+	147	12000+	12062.8
4	388	5002.8+	20007.6+	3525	12000+	12000+
5	399	1000.18+	20008.9+	3492	12000+	12000+
6	5000+	1003.34+	20008.3+	3461	12000+	12000+
7	4233.1	1000.66+	20009.2+	3518	12000+	12000+
8	477	5009.88+	24019+	3608	12000+	12000+
9	644	5026.7+	20016.4+	4111.8+	12000+	12000+
10	24	389	635	566	1860	3129
11	23	473	1634	3710	12000+	4799
12	22	419	1746	1963	9093	3494
13	60	373	18484.4	126.2	1531	9442
14	61	364	850	126.2	1364	4034
15	26	425	1321	136.2	3773	9255
16	27	388	2433	127.2	3286	2198
17	66	365	1115	129.2	404	10971
18	61	444	823	352.2	2413	3092
19	25	277	644	726	2447	1644
20	23	325	1315	471	1009	3105
21	24	330	1199	765	11031	3332
22	34	292	20015.7+	905	8305	9449
23	38	267	1261	924	1071	7651
24	27	407	1034	500	750	2180
25	37	547	576	528	2935	7597
26	41	293	1327	548	6583	6452
27	26	324	596	2575	1637	3532

Table 5.9: Summary result of the performance of the families of test problems TP3 and TP4.

Exp	TP3(2)	TP3(4)	TP3(6)	TP4(2)	TP4(4)	TP4(6)
1	727	1005+	2000+	267	536	872
2	543	1002.2+	2018.8+	282	580	1090
3	327.8	5039.7+	2010.6+	147	700	1356
4	409	5031+	2010.5+	285	549	893
5	348	5013.3+	2011.1+	274	556	1230
6	1080+	5008.4+	2018+	272	723	1081
7	233	5009.4+	2010.6+	287	576	947
8	464	5019.2+	2012.3+	293	635	937
9	711	1001.2+	2008+	1111.8+	577	1294
10	25	413	796	426	1000+	807
11	22	395	1730	729	1000+	760
12	23	376	720	1022.3	1000+	2000+
13	59	417	1206	126.2	1000+	739
14	62	357	1238	126.2	839	2000+
15	26	269	1192	136.2	1000+	2000+
16	60	417	1069	135.2	780	2000+
17	32	381	746	185	1000+	643
18	61	420	1806	1000+	412	2000+
19	25	600	568	226	775	772
20	23	267	1854	234	501	2000+
21	23	270	480	1000	1000+	2000+
22	33	486	900	285	1000+	2000+
23	38	316	563	192	1000+	2000+
24	27	312	695	1000+	1000+	2000+
25	27	351	641	212	1000+	2000+
26	32	283	603	203	1000+	789
27	26	386	642	1000+	318	2000+

Table 5.10: Summary result of the performance of the families of test problems TP5 and TP6.

Exp	TP5(2)	TP5(4)	TP5(6)	TP6(2)	TP6(4)	TP6(6)
1	426	1086.6	7321.4	398	1086.6	3321.4
2	537	901.7	5484.8	389	901.7	1484.8
3	272	1039.1	1607.2	302	1009.1	1607.2
4	274	3706	2028.7	206	3218	2477.07
5	549	1664.4	3407.3	404	1664.4	3407.29
6	690	1656.5	1487.0	409	1656.5	1486.99
7	311.34	26000.04	19889.5	204	1695	1731.8
8	483	1779.7	4968.7	280	1779.7	4968.6
9	552.5	2065	1468.8	463	2065.1	1468.8
10	234.5	1233	5298.2	350	1218	5298.2
11	216.5	1682	2888.6	318	1558	2888.6
12	226.5	1402	2554.8	305	1782	2554.76
13	288.5	25742.85	6124.0	378	1145	6124.05
14	256.5	1601	5547.5	363	1567	5547.54
15	235.5	2129	2911.2	338	2272	2911.2
16	198	7723.47	2505.2	175	2494	2505.17
17	202.5	1372	6008.0	336	1851	4798.66
18	238.5	1535	2737.3	303	1885	2737.28
19	201.5	1454	1923.0	330	1368	7777
20	245.5	1298	3294.9	350	1098	3294.9
21	220.5	1244	3736.5	334	1260	2676.07
22	267.5	11388.595	3378.3	351	1006	3378.3
23	246.5	1120	2391.5	352	1172	2638.11
24	227.5	1396	2912.3	338	1667	2912.27
25	182	1344	19878.9	218	1288	2969.57
26	198.5	1329	2657.9	365	1649	2657.93
27	229.5	1212	3104.0	331	1025	2891.57

Table 5.11: Summary result of the performance of the families of test problems TP7(4), TP8(4), TP9(4) and TP10(4).

Exp	TP7(4)	TP8(4)	TP9(4)	TP10(4)
1	6023.6+	756.69	2000.75+	3381
2	6000+	646.69	2007.11+	2636
3	5752.2	736	2001.6+	3395
4	650	534	2005+	3125
5	6000+	524	6016.1+	2754
6	10018.3+	580.69	6007.1+	3134
7	6023.95+	571.69	1386.2	2798
8	6016.3+	809	6007.2+	3396
9	10027.4+	819.69	2010.7+	2874
10	577	389.69	455	1199
11	637	396	462	1629
12	527	378	521	2423
13	687.6	579.69	263	1799
14	554	394	450	1756
15	839.6	414.69	655	2174
16	444	346	529	1741
17	514.6	496.69	646	2095
18	514.6	444.69	418	373.38
19	441	459.69	289	1809
20	372	422	377	2050
21	377	460	343	2640
22	663	475	290	1775
23	397	438	549	1316
24	461.6	546.69	313	1560
25	392	403	568	1270
26	424.6	580.69	367	1627
27	498.6	476.69	342	1718

Case study 2: non-obtuse solid angle feasible cone

Summary of the PM for the case study non-obtuse solid angle feasible cone are shown from Table 5.12 to Table 5.15.

Table 5.12: Summary result of the performance of the families of test problems TP11 and TP12.

Exp	TP11(2)	TP11(4)	TP11(6)	TP12(2)	TP12(4)	TP12(6)
1	289	5500.3+	9505.46+	3573	8514	17255
2	4290	5509.1+	9523.58+	3483	9670	19539
3	4357	5536.9+	9513.91+	3645	9689	24125
4	4313	5502.6+	9506.6+	3500	9873	17123
5	4313	5502.6+	9521.9+	3567	9207	20036
6	400+	5532.1+	9527.8+	3600	9897	23817
7	4313	5501.71+	9503.69+	3564	8629	17107
8	4313	5520.8+	9505.88+	3575	9468	19107
9	400+	5511.9 +	9537.16+	3628.2	9826	25519
10	22	301	1205	1479	4427	2579
11	22	483	1556	2190	936	3577
12	21	254	2068	122.9	6334	2240
13	81.75	332	1229	124.9	2476	16437
14	39	281	921	129.9	2722	1394
15	101.9	359	1138	125.9	9373	3707
16	81.75	255	3842	127.9	1820	6184
17	39	255	698	164.9	7577	3097
18	101.5	344	5443	162.9	847	3713
19	24	504	4776	518	2535	13248
20	22	326	1529	3077	634	1722
21	22	231	707	658	2350	1745
22	31	1329	1240	2887	1546	4623
23	32	1500	1878	3260	1021	3729
24	29	437	1419	107.9	978	3091
25	31	371	1243	109.9	1012	7169
26	32	345	2516	345	1843	1850
27	29	344	1999	96	1798	1739

Table 5.13: Summary result of the performance of the families of test problems TP13 and TP14.

Exp	TP13(2)	TP13(4)	TP13(6)	TP14(2)	TP14(4)	TP14(6)
1	290	4701.87+	13102.7+	307	567	1444
2	290	4702.74+	13224.4+	278	625	8189
3	357	4737.6+	13588.3+	445+	672	1514
4	314	700+	13103.78+	290	667	1099
5	314	4702.3+	19820+	279	669	3706
6	400+	4798.6+	13190.4+	400	813	7898.5+
7	314	700+	5337.12	310	674	1341.5
8	314	4767.4+	5344.12	321	626	1348.5
9	400+	4715.4+	5642.12	428.2+	627	1646.5
10	23	263	570.06	400+	1100+	3800+
11	22	195	576	428.2+	1022	1749
12	20	361	874	123.9	1107.65+	3800+
13	81.7	249	9017.06	124.9	1107.56	3800+
14	39	200	1555	123.9	893	1943
15	102.4	365	4892	128.9	1107.56+	3800+
16	81.8	327	1926	127.9	1100	3800+
17	38	252	1672	164.9	800	3800+
18	102.4	307	9109.43+	135.9	1112.6+	3800+
19	24	304	1036	317	1100.4	3806.2+
20	23	267	9100+	310	464	1528
21	23	444	668.42	105.9	1107.56+	1514
22	32	700	5992.43	306	1106.83+	1099
23	32	608	2553	363	541	3806.2+
24	30	465	1390	105.9	1107.56+	3806.7+
25	32	401	9100.528+	108.9	1108.26+	816
26	32	400	1710	408.9+	874	3806.1+
27	31	381	1288	103.9	1112.6+	3800.49+

Table 5.14: Summary result of the performance of the families of test problems TP15 and TP16.

Exp	TP15(2)	TP15(4)	TP15(6)	TP16(2)	TP16(4)	TP16(6)
1	453	1164.3	2061	206	2850	2061
2	314	7274.1+	23295.6+	273	22594.9+	10764.2+
3	503.54+	790.1+	3310.5	216	790.1	4172.9
4	4617.8	7390+	2428.8	260	22569.9+	2321.1
5	4622.8	6805.54+	2004.3	317	1604	1845.9
6	4622.8	7333.8+	1056.8	340	2422	2969.6
7	4617.8	7441.2+	19629.1+	4649.8	4492.7	2194.6
8	4622.8	7085.8+	2309	4649.8	22197.79+	2309
9	4622.8	7448.9+	3570.5	4649.8	22616.66+	4086.7
10	49	796	3307	189.9	766.6	3274.3
11	48	668.2	2512	195.9	646	6844.3
12	50	935	5728.6	64.4	927	4494.1
13	48	2343	19025.4+	185.9	2254	3897.1
14	48	2802.7	3580.2	193.9	1604	5834.5
15	19	1326.7	4033.2	195.9	2422	2969.6
16	48	1280.7	6170.3	186.9	4492.7	4157.6
17	48	1083.2	4454.2	193.9	17989.7	3118.3
18	19	706	3251.6	193.9	543.7	4086.7
19	23	1150	13275.4	165.9	861	3191.8
20	23	770	1774.4	117.9	2158	2283.3
21	23	1151.2	2522.8	172.9	1297	4510.4
22	20	2815.8	18740.2	180	898	4914.3
23	20	779.2	2696	180	746.7	4529
24	19	644.7	2463.3	145	1383	2645.3
25	20	1361	4602	256	1163	1846.2
26	20	839.2	4328.8	113.9	1400	4756.5
27	19	1303	2919	240	842	2834

Table 5.15: Summary result of the performance of the families of test problems TP17(4), TP18(4), TP19(4) and TP20(4).

Exp	TP17(4)	TP18(4)	TP19(4)	TP20(4)
1	1252	5576.6+	4420+	2748
2	23540+	5247.4+	4401.29+	8164.4+
3	23081.2+	5158+	4950.3+	7733.3+
4	23013.6+	679	4850.36+	3095
5	23137.9+	5456+	4408.895+	3124
6	23466.7+	5502.1+	4771.57+	8091.2+
7	23360+	594.7	4400.652+	7708.3+
8	23398.1+	679.7	4842.1+	7709.8+
9	23038.5+	492.7	4821.3+	2969
10	4860.9	504.7	279	2013
11	717	1098.8	621	350.9
12	280.3	334.6	172	293.4
13	18650.9	632.7	334	347.9
14	241	750.7	277	260.46
15	500	783	166	280
16	19020+	397.7	171	2025
17	377	442	165	3068
18	259	553.7	336	2969
19	517	571.7	309	1799
20	535.9	575.7	289	307.9
21	537	450.7	261	2800
22	367	678.7	525	310.9
23	527	422	284	3632
24	547	605.7	267	388.9
25	429.9	411.7	300	1924
26	418.9	427.7	317	1944
27	349	995.7	358	378.9

5.5 Conclusions

As can be appreciated from the numerical test problems, both the LCNM method and the LCNM+PC method had better performance than the SM. In most test problems, our methods converged at least to a feasible local minimum within a good precision, regarding the NE. In contrast, the SM converged to points that not necessarily are local optima of the test problems, and in some cases it converged to infeasible points. Moreover, the accuracy of the SM was deficient in many test problems.

Hence, the LCNM method and its premature collapse mode present advantages over the SM, even though the LCNM method and the LCNM+PC method were relatively expensive for some settings of the reflection coefficient α and step size parameter τ. However, our methods converged to a feasible point with an adequate accuracy within a low NE, when the reflection coefficient α takes values of 0.95 or 1, and the step size parameter τ is equal to 1 for the most of the test problems.

Analysing the best average PM reported in Section 5.4, we can also observe that the LCNM+PC method had better performance than the LCNM method, for the cases of dimension 4 and 6. In contrast, for the 2-dimensional test problems, the LCNM method evidenced advantages over the LCNM+PC method.

Furthermore, the LCNM method identified constrained optima with lower NE than its premature collapse version, when the solid angle is non-obtuse. Whereas, the LCNM+PC method obtained better performance than the LCNM method for the case of dimension 4 and obtuse solid angle only, conforming with the analysis of the maximum values of the PM's.

This new procedure improves the performance of the LCNM method when the constrained optimization problem has a few number of linear constraints. However, if the constrained minimization problem has a large number of linear constraints, the modified method is not significantly better.

In summary, based on the analysis of the maximum value of the PM's we can observe that the LCNM method shows advantages over the LCNM+PC method. However, the LCNM+PC method offers better accuracy than the LCNM method.

Chapter 6

Exploring other variations

6.1 Introduction

The study of the LCNM algorithm and its premature collapse version has moved on a new explorations for improving the efficiency of the LCNM algorithm. This has motivated the assessment of two new variations called *dynamic LCNM algorithm* and *directional NM algorithm* that were considered to have potential. Both approaches compute the reflection trial point by taking into account the best direction of optimization. Numerical tests have shown that these variations can be more expensive than the original LCNM algorithm.

Nevertheless, this chapter describes both variations because their study could help us to understand the behaviour of the methods based on the simplex.

The content of this chapter is as follows. Section 6.2 describes the dynamic LCNM algorithm whose principle is the estimation of the best value of expansion coefficient γ and contraction coefficient β for optimizing the objective function in the direction $\mathbf{x}_{\max}$ towards $\mathbf{x}_{cen}$. Numerical examples are shown for comparing the dynamic LCNM algorithm against our original version. The development of the directional NM algorithm is presented in Section 6.3. An initial study of this variation of the NM algorithm is made, to evaluate its advantage for being incorporated in the LCNM algorithm. We also present numerical examples for comparing the NM algorithm versus this directional NM version. Finally, in Section 6.4 we give some conclusions about these new variations.

6.2 The dynamic LCNM algorithm

We shall herein propose a variant of the LCNM algorithm based on selecting values of expansion coefficient and contraction coefficient, by minimizing an objective function that is defined by the points $[f(\mathbf{x}_{\max}),\mathbf{x}_{\max}^T]^T$, $[f(\mathbf{x}_{cen}),\mathbf{x}_{cen}^T]^T$ and $[f(\mathbf{x}_{refl}),\mathbf{x}_{refl}^T]^T$.

To study this new approach, we shall consider the following propositions for the NM algorithm.

Proposition 6.1 (Optimum θ) *Let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a non-linear continuous objective function of an unconstrained minimization problem. Let $S^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_v]$ be a qth sorted entire simplex defined in the Euclidean space $\mathbb{R}^d$, that is, the qth simplex is formed by $(d+1)$ sorted vertices conforming with f_i, where f_i represents the value of $f(\mathbf{x}_i)$ for all $i = 1, \ldots, v$. Let $\mathbf{x}_{cen}$ be the centroid of the remaining hyperface of the current qth simplex given by*

$$\mathbf{x}_{cen} = \frac{1}{v-1}\sum_{i=1}^{v-1}\mathbf{x}_i$$

Let $\mathbf{x}_{refl}$ be the reflection trial point which is computed by

$$\mathbf{x}_{refl} = (1+\alpha)\mathbf{x}_{cen} - \alpha\mathbf{x}_v \quad \forall\alpha > 0$$

and whose value is computed by the objective function $f(\mathbf{x})$.

Let

$$\mathbf{x}_\theta = (1+\theta)\mathbf{x}_{cen} - \theta\mathbf{x}_v, \tag{6.1}$$

be the points located on the straight line $l_{v,cen}$ defined by the points $\mathbf{x}_v$ and $\mathbf{x}_{cen}$ for all $-\infty < \theta < \infty$. Let $f(\theta) : \mathbb{R} \to \mathbb{R}$ be an unknown parametric function, which represents the objective function on the straight line $l_{v,cen}$, and uniquely depends on θ for all $-\infty < \theta < \infty$.

If the function $f(\theta)$ can be approximated to a second degree polynomial $g(\theta) : \mathbb{R} \to \mathbb{R}$ given by

$$f(\theta) \approx g(\theta) = c_0 + c_1\theta + c_2\theta^2, \tag{6.2}$$

and $c_2 > 0$, then a minimum of $f(\mathbf{x})$ is approximately located either at

$$\mathbf{x}_{\theta^*} = (1+\theta^*)\mathbf{x}_{cen} - \theta^*\mathbf{x}_v, \tag{6.3}$$

where

$$\theta^* = \frac{f_{refl} + (\alpha^2 - 1) f_{cen} - \alpha^2 f_v}{2(\alpha + 1) f_{cen} - 2 f_{refl} - 2\alpha f_v}, \tag{6.4}$$

otherwise, its minimum is obtained as $\theta \to -\infty$ or $\theta \to \infty$.

Proof. Let $g(\theta)$ be the second degree polynomial defined by the points $[f_v, \mathbf{x}_v^T]^T$, $[f_{cen}, \mathbf{x}_{cen}^T]^T$ and $[f_{refl}, \mathbf{x}_{refl}^T]^T$. By Equation (6.1) and Equation (6.2), we obtain

$$\begin{bmatrix} f_v \\ f_{cen} \\ f_{refl} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 \\ 1 & 0 & 0 \\ 1 & \alpha & \alpha^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix}, \tag{6.5}$$

whose vector $\mathbf{c} = [c_0, c_1, c_2]^T$ is computed by

$$\begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\frac{\alpha}{\alpha+1} & \frac{1}{\alpha}(\alpha - 1) & \frac{1}{\alpha+\alpha^2} \\ \frac{1}{\alpha+1} & -\frac{1}{\alpha} & \frac{1}{\alpha+\alpha^2} \end{bmatrix} \begin{bmatrix} f_v \\ f_{cen} \\ f_{refl} \end{bmatrix} = \begin{bmatrix} f_{cen} \\ \frac{\alpha-1}{\alpha} f_{cen} - \frac{\alpha}{\alpha+1} f_v + \frac{1}{\alpha+\alpha^2} f_{refl} \\ \frac{1}{\alpha+1} f_v - \frac{1}{\alpha} f_{cen} + \frac{1}{\alpha+\alpha^2} f_{refl} \end{bmatrix} \tag{6.6}$$

By differentiating $g(\theta)$ with respect to θ and setting the derivative equal to 0, we obtain the stationary point of the second degree polynomial

$$\theta^* = -\frac{1}{2}\frac{c_1}{c_2} = \frac{f_{refl} + (\alpha^2 - 1) f_{cen} - \alpha^2 f_v}{2(\alpha + 1) f_{cen} - 2 f_{refl} - 2\alpha f_v},$$

which corresponds to a minimum if $c_2 > 0$.

If $c_1 \neq 0$ and $c_2 = 0$, then the minimum of $g(\theta) = c_0 + c_1\theta$ is as $\theta \to -\infty$ or $\theta \to \infty$, which evidently depends on the sign of c_1. On the other hand, if $c_2 < 0$, then θ^* corresponds to a maximum, therefore $g(\theta)$ has two minima at both $\mathbf{x}_\theta$ as $\theta \to -\infty$ and $\theta \to \infty$. ∎

The value of θ^* can be employed by the NM algorithm for attempting an expansion step or a contraction step, in order to estimate either the best expansion coefficient γ or contraction coefficient β at each iteration of the NM algorithm. The accuracy obviously depending on the fitting of $g(\theta)$ to the function $f(\boldsymbol{\theta})$. Since the function $g(\theta)$ can have at least a minimum as $\theta \to -\infty$ or $\theta \to \infty$, if $c_2 \leq 0$, it is necessary to constrain the value of θ to an interval for avoiding a collapse of the current simplex on its own remaining hyperface when $\theta^* = 0$, or its collapse when $\theta \to -\infty$ or $\theta \to \infty$.

Due to this fact, we enunciate the following propositions:

Proposition 6.2 (Constrained optimum θ) *Let $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ be a non-linear continuous objective function of an unconstrained minimization problem. Let $g(\theta)$ be the second degree polynomial given by Equation (6.2), which is approximately equal to $f(\theta)$. If the NM algorithm is being applied to the minimization problem and at the qth iteration the algorithm is an expansion step or contraction step under the condition that $f_{refl} < f_1$ or $f_2 < f_{refl}$ respectively, then the solution θ_c^* to the problem of minimizing $g(\theta)$ subject to $L \leq \theta \leq U$ is given by*

$$\theta_c^* = \begin{cases} \max[L, \min(\theta^*, U)] & \text{if } c_2 > 0, \\ \arg\left[\min\left[g(L), g(U)\right]\right] & \text{otherwise} \end{cases}, \tag{6.7}$$

where $0 < L < U$ if the NM algorithm is at an expansion step or $-1 < L < U < 0$ if the NM algorithm is at an inside contraction step.

Proof. We assume $f(\theta)$ is approximately equal to the continuous function $g(\theta)$. There exists several cases depending on the sign of c_2 and the value of the unconstrained optimum θ^* given by Proposition 6.1. We have

a) If $c_2 = 0$, then the constrained minimum of $g(\theta)$ can be located at L or U.

b) If $c_2 < 0$ and $\theta^* \leq L$, then the constrained minimum of $g(\theta)$ is at U.

c) If $c_2 < 0$ and $\theta^* \geq U$, then the constrained minimum of $g(\theta)$ is at L.

d) If $c_2 < 0$ and $L < \theta^* < U$, then the constrained minimum of $g(\theta)$ can be located at L, U or both.

e) If $c_2 > 0$ and $\theta^* \leq L$, then the constrained minimum of $g(\theta)$ is at L.

f) If $c_2 > 0$ and $\theta^* \geq U$, then the constrained minimum of $g(\theta)$ is at U.

g) If $c_2 > 0$ and $L < \theta^* < U$, then the constrained minimum of $g(\theta)$ is at θ^*. ■

6.2.1 The modified algorithm

Since the convergence of the vertices of the simplex to a point is often carried out by contraction operations and occasionally by shrinkage operations, the approach of estimating a constrained minimum parameter θ_c^* was implemented for estimating its optimum during the expansion operation, because its cost involves two function evaluations per iteration of the modified NM algorithm. Nonetheless, the implementation for estimating the constrained optimum contraction value was also done for corroborating the cost of employing this approach in contraction operations.

Based on Proposition 6.1 and Proposition 6.2, two versions of the LCNM algorithm were implemented, named the $\hat{\gamma}$-LCNM algorithm and the $\hat{\beta}\hat{\gamma}$-LCNM algorithm, for identifying the LCNM algorithm when a constrained minimum expansion coefficient γ is optimized and, when a constrained optimization of both parameters β and γ is carried out, respectively.

As a result of this viewpoint, we propose the following modification to steps 6 and 7 of the LCNM algorithm, depending on the mode of operation.

Step 6. Attempt Constrained Expansion based on optimum expansion coefficient

if $f_{refl} \leq f_{\min}$

then Given $1 \leq L_\gamma \leq \theta \leq U_\gamma$, estimate the optimum value of θ_c^* by Proposition 6.2

Compute $\mathbf{x}_{\exp} = (1 + \theta_c^*)\mathbf{x}_{cen} - \theta_c^*\mathbf{x}_{\max}$

Perform **Linear constraint procedure** to $\mathbf{x}_{\exp}$ considering that

$$\mathbf{x}_o = \mathbf{x}_{refl} \text{ and } \mathbf{x}_{new} = \mathbf{x}_{\exp}.$$

if $f_{\exp} \leq f_{\min}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{\exp}$, $f_{\max} \leftarrow f_{\exp}$

else $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$

Go to step 10

else go to step 7

Step 7. Attempt Contraction based on optimum contraction coefficient

if $f_{refl} > f_{ntw}$

then Reduce the size of the current simplex either by a contraction or by shrinking, which we shall need to verify the following.

if $f_{refl} \leq f_{\max}$

then $\mathbf{x}_{\max} \leftarrow \mathbf{x}_{refl}$, $f_{\max} \leftarrow f_{refl}$

Sort $f_j \quad \forall \quad j = 1, 2, \ldots, v$ for determining $\mathbf{x}_{\min}$, $\mathbf{x}_{ntw}$ and $\mathbf{x}_{\max}$.

Given $-1 < L_\beta \leq \theta \leq U_\beta < 0$, estimate the optimum value of θ_c^* by Proposition 6.2

Compute the contraction point $\mathbf{x}_{cont} = (1 + \theta_c^*)\mathbf{x}_{cen} - \theta_c^*\mathbf{x}_{\max}$ and its value f_{cont} and go to step 8.

Observe that the $\hat{\beta}\hat{\gamma}$-LCNM algorithm requires the modified step 6 and 7 of the LCNM algorithm, whilst the $\hat{\gamma}$-LCNM algorithm only needs the modification of the step 6.

6.2.2 The LCNM algorithm vs the dynamic LCNM algorithms

We shall herein show some preliminary numerical examples for comparing the LCNM algorithm and its dynamic version through some test problems of Chapter 5. For this, we employ the following settings for carrying out both algorithms in each numerical test problem.

The parameters used by the LCNM method were $\alpha = 1$, $\beta = 0.5$, $\gamma = 2$ and $\delta = 0.5$. The parameters for the dynamic LCNM algorithm were: $\alpha = 1$, $0.4 \leq \beta \leq 0.6$, $1 \leq \gamma \leq 2$ and $\delta = 0.5$. The initial point for all test problems was the point $(20, 20, 20, 20)^T$.

Furthermore, the parameters of the stopping rule were $\eta_1 = \eta_2 = 10^{-6}$ and the step size parameter τ was fixed to 1, for all algorithms.

Table 6.1 displays numerical results reported by the LCNM algorithm, the $\hat{\gamma}$-LCNM algorithm and the $\hat{\beta}\hat{\gamma}$-LCNM algorithm, using the performance measure, denoted by PM

$$PM_i = NE_i + 1000\ DTP_i \ \ \forall i$$

Note that in 13 of 16 test problems the PM of the $\hat{\gamma}$-LCNM algorithm is less than the PM of the $\hat{\beta}\hat{\gamma}$-LCNM algorithm, which corroborates the fact that the $\hat{\beta}\hat{\gamma}$-LCNM algorithm is the more expensive, due to contraction operations that are carried out during the convergence of the simplex to a point.

Table 6.1: Comparison between the LCNM and the dynamic LCNM algorithms

	LCNM			$\hat{\gamma}$ LCNM			$\hat{\beta}\hat{\gamma}$ LCNM		
	PM	NE	DTP	PM	NE	DTP	PM	NE	DTP
TP1(4)	428	428	1.17E-06	398	398	1.60E-06	611	611	2.00E-06
TP2(4)	4488	4488	4.84E-73	5643	5643	1.0E-162	7566	7566	1.0E-162
TP5(4)	67715	935	6.68E+01	1938	1938	2.13E-07	46540	1907	4.46E+01
TP6(4)	67715	935	6.68E+01	2230	2230	8.71E-07	46540	1907	4.46E+01
TP7(4)	848	848	6.06E-07	538	538	9.64E-07	972	972	4.46E-07
TP8(4)	603	603	6.78E-07	2397	427	1.97E+00	2517	547	1.97E+00
TP9(4)	623	623	5.03E-06	483	483	5.12E-06	594	594	5.16E-06
TP10(4)	1565	1565	3.78E-15	1865	1865	1.20E-17	2174	2174	1.76E-16
TP11(4)	359	359	4.62E-07	339	339	4.29E-07	362	362	3.18E-07
TP12(4)	9373	9373	3.70E-85	6177	6177	1.0E-162	7666	7666	1.0E-162
TP15(4)	6843	1271	5.57E+00	44870	44870	1.11E-07	2545	2545	1.32E-07
TP16(4)	2422	2422	1.21E-07	25631	18355	7.28E+00	3085	3085	7.76E-07
TP17(4)	500	500	7.08E-07	379	379	1.09E-05	976	976	4.64E-07
TP18(4)	783	783	8.82E-07	2451	481	1.97E+00	2670	487	2.18E+00
TP19(4)	1080	166	9.14E-01	1072	158	9.14E-01	1333	419	9.14E-01
TP20(4)	1574	267	1.31E+00	1871	1871	3.20E-16	1422	475	9.47E-01

Table 6.2 (on page 168) depicts the percentage change of the $\hat{\gamma}$-LCNM algorithm and the $\hat{\beta}\hat{\gamma}$-LCNM algorithm with respect to the LCNM algorithm, calculated using the following equation.

$$\delta PM_{dyn} = 100 \cdot \frac{PM_{LCNM} - PM_{dyn}}{PM_{LCNM}}$$

Table 6.2: Percentage of the dynamic LCNM algorithms

	$\delta PM_{\hat{\gamma}\text{-LCNM}}$	$\delta PM_{\hat{\beta}\hat{\gamma}\text{-LCNM}}$
TP1(4)	7.0	-42.8
TP2(4)	-25.7	-68.6
TP5(4)	97.1	31.3
TP6(4)	96.7	31.3
TP7(4)	36.6	-14.6
TP8(4)	-297.5	-317.4
TP9(4)	22.5	4.7
TP10(4)	-19.2	-38.9

	$\delta PM_{\hat{\gamma}\text{-LCNM}}$	$\delta PM_{\hat{\beta}\hat{\gamma}\text{-LCNM}}$
TP11(4)	5.6	-0.8
TP12(4)	34.1	18.2
TP15(4)	-555.7	62.8
TP16(4)	-958.3	-27.4
TP17(4)	24.2	-95.2
TP18(4)	-213.0	-241.0
TP19(4)	0.7	-23.4
TP20(4)	-18.9	9.7

As can be seen from Table 6.2, the $\hat{\gamma}$-LCNM algorithm has an improvement in 9 of the 16 test problems (positive value of $\delta PM_{\hat{\gamma}\text{-LCNM}}$) with respect to the LCNM algorithm, which does not represent a significant advantage of the $\hat{\gamma}$-LCNM algorithm over the LCNM algorithm. Moreover, in 6 of the 16 test problems the $\hat{\gamma}$-LCNM algorithm has a percentage of change more than 8%. Despite the limited comparative experiments carried out, the $\hat{\gamma}$-LCNM algorithm does not seem to be a method more efficient than the LCNM algorithm, due to the fact that the $\hat{\gamma}$-LCNM algorithm reported percentage of changes less than -200% in 4 test problems and its non-significant advantage over the LCNM algorithm.

Note that the $\hat{\gamma}$-LCNM algorithm requires two function evaluations per expansion operation, whilst the LCNM algorithm evaluates the objective function once per expansion operation. That can be a reason for what the dynamic LCNM algorithms are not good enough in comparison with the LCNM algorithm.

In addition, we believe that the $\hat{\gamma}$-LCNM algorithm can have advantages over the LCNM algorithm when the objective function is convex, because in 3 of the 4 convex test problems the $\hat{\gamma}$-LCNM algorithm reported better performance than the LCNM algorithm.

Nonetheless, the study of other algorithms based on the dynamic LCNM algorithms might be an interesting investigation, where other modifications for the LCNM algorithm can result in an improvement of the LCNM algorithm.

6.3 A variation of the NM based on descent direction

In this section, we study other operations of the simplex for obtaining other trial points, using approaches principally based on a descent direction that can be estimated by either the gradient of the objective function hyperplane (GOFH) or the descent direction of the simplex edges (DDSE).

Suppose the unconstrained minimization problems given by

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}),$$

where $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ is a non-linear function.

For the class of problems given above, the estimation of a descent direction via GOFH is obtained by calculating the gradient of the hyperplane defined by the points on the surface of the objective function and whose projections into the domain set are just the vertices of the current simplex. That is shown in Figure 6.1, where the points $[f_i : \mathbf{x}_i^T]^T \in \mathbb{R}^3 \; \forall i = 1, 2, 3$ are on the surface of the objective function, which define the plane displays in the figure.

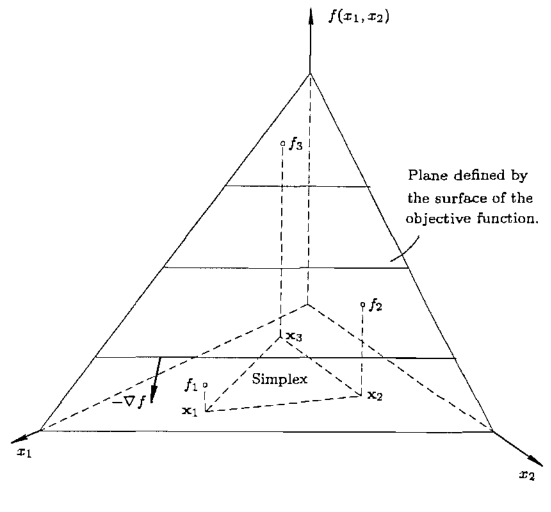


Figure 6.1: Estimation of a descent direction via GOFH

In the case of the DDSE, the descent direction is defined by the average vector of the vectors given by the edges of the current simplex. Figure 6.2 illustrates an example in two dimension, where a descent direction is defined by the edge vectors $\mathbf{x}_1 - \mathbf{x}_2$ and $\mathbf{x}_1 - \mathbf{x}_3$, so the qth descent direction vector $\mathbf{d}^{[q]}$ is computed by a function of those edge vectors.

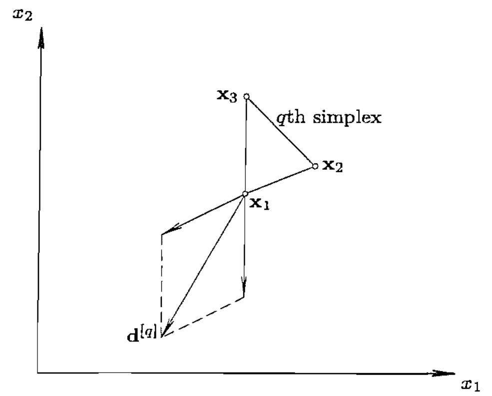


Figure 6.2: Estimation of a descent direction via DDSE

Both approaches look like a good ideas, because the identification of a direction of descent could improve the efficiency of the NM method. However, theoretical explanation and numerical examples show that this variant of the NM method is very sensitive to the value of the objective function, so such methods can become vulnerable, especially if the objective function is affected by noise.

6.3.1 The directional Nelder-Mead method

To study the variant of the NM method named the directional NM method, we shall enunciate the following propositions.

Proposition 6.3 (Descent direction) *Let $S^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \cdots : \mathbf{x}_v]$ be a qth entire simplex defined in the Euclidean space $\mathbb{R}^d$, that is, the qth simplex is formed by $(d+1)$ vertices and whose matrix defined by $E^{[q]} = [\mathbf{x}_2 - \mathbf{x}_1 : \mathbf{x}_3 - \mathbf{x}_1 : \cdots : \mathbf{x}_v - \mathbf{x}_1]$ represents the edges of the simplex that intersect at the common vertex $\mathbf{x}_1$, and whose column rank is equal to d. Let $f(\mathbf{x})$ be the objective function of the minimization problem, and let f_i denote the value of $f(\mathbf{x}_i)$ for all $i = 1, \ldots, v$. Then* *a) The normalized direction in the Euclidean space $\mathbb{R}^d$ given by*

$$\mathbf{d}^{[q]} = -\frac{\boldsymbol{\beta}_x^{[q]}}{\left\|\boldsymbol{\beta}_x^{[q]}\right\|} \tag{6.8}$$

is a descent direction of the objective function,

where $[\beta_0^{[q]} : \boldsymbol{\beta}_x^{[q]}]^T = [\beta_0^{[q]}, \beta_1^{[q]}, \beta_2^{[q]}, \dots \beta_v^{[q]}]^T$ is the vector of coefficients of the hyperplane defined by the points

$$[f(\mathbf{x}_i) : \mathbf{x}_i^T]^T \in \mathbb{R}^{d+1} \ \forall i = 1, 2, \dots, v.$$

b) The normalized direction in the Euclidean space $\mathbb{R}^d$ given by

$$\mathbf{d}^{[q]} = \frac{\sum_{i=2}^{v} (f_i - f_1) \frac{(\mathbf{x}_1 - \mathbf{x}_i)}{\|\mathbf{x}_1 - \mathbf{x}_i\|}}{\left\|\sum_{i=2}^{v} (f_i - f_1) \frac{(\mathbf{x}_1 - \mathbf{x}_i)}{\|\mathbf{x}_1 - \mathbf{x}_i\|}\right\|} \tag{6.9}$$

is a descent direction of the objective function.

Proof. Part a)

Let $h(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a hyperplane in the Euclidean space $\mathbb{R}^{d+1}$ that is defined by the vertices of the simplex, thus

$$h(\mathbf{x}) = X\boldsymbol{\beta}^{[q]} + \epsilon$$

Since the current qth simplex is entire in the Euclidean space $\mathbb{R}^d$, the number of vertices (v) of the current qth simplex $S^{[q]}$ is $d+1$, which satisfies the minimum necessary number of points that must define the hyperplane $h(\mathbf{x})$ that contain the points $[f(\mathbf{x}_i) : \mathbf{x}_i^T]^T \in \mathbb{R}^{d+1} \ \forall i = 1, \dots, v$. Hence, all points $[f(\mathbf{x}_i) : \mathbf{x}_i^T]^T \in \mathbb{R}^{d+1} \ \forall i = 1, \dots, v$ perfectly fit to the hyperplane with error zero. Therefore,

$$h(\mathbf{x}) = X\boldsymbol{\beta}^{[q]} = \beta_0^{[q]} + \sum_{i=1}^{d} \beta_i^{[q]} x_i, \tag{6.10}$$

where the vector of coefficients $\boldsymbol{\beta}^{[q]} = [\beta_0^{[q]} : \boldsymbol{\beta}_x^{[q]}]^T = [\beta_0^{[q]}, \beta_1^{[q]}, \beta_2^{[q]}, \dots \beta_v^{[q]}]^T$ can be computed by

$$\boldsymbol{\beta}^{[q]} = X^{-1}\mathbf{f}^{[q]}. \tag{6.11}$$

Here the matrix $X = \begin{bmatrix} 1_v : & S^{[q]^T} \end{bmatrix}$ and the column vector $\mathbf{f}^{[q]} = [f_1, \ f_2, \dots, f_v]^T$ is formed by the values of the objective function at each vertex of the qth current simplex

$S^{[q]}$. Therefore, a normalized descent direction can be defined by

$$\mathbf{d}^{[q]} = -\frac{\nabla h_x(\mathbf{x})}{\|\nabla h_x(\mathbf{x})\|} = -\frac{\beta_x^{[q]}}{\left\|\beta_x^{[q]}\right\|}$$

Part b)

Since $f_1 \leq f_2 \leq \cdots \leq f_v$, then $f_1 \leq f_i$ for all $i = 2, \ldots, v$, and $(\mathbf{x}_1 - \mathbf{x}_i)$ is a non-ascendant direction along the edge defined by the points $\mathbf{x}_1$ and $\mathbf{x}_i$ $\forall i = 2, \ldots, v$. In the case that $f_1 < f_i$ for some $i \in \{i \in \mathbb{N} \mid 2, \ldots, v\}$, the non-zero vector given by $(\mathbf{x}_1 - \mathbf{x}_i)$ is a descent direction along the edge from $\mathbf{x}_i$ to $\mathbf{x}_1$. Thus, for all $i = 2, \ldots, v$,

$$(f_i - f_1)\frac{(\mathbf{x}_1 - \mathbf{x}_i)}{\|\mathbf{x}_1 - \mathbf{x}_i\|} = \begin{cases} \Delta_i \frac{(\mathbf{x}_1 - \mathbf{x}_i)}{\|\mathbf{x}_1 - \mathbf{x}_i\|} & \text{is a non-zero descent vector} \quad \text{If } f_1 < f_i, \\ \mathbf{0}_d & \text{is a zero vector} \quad \text{If } f_1 = f_i \end{cases} \tag{6.12}$$

where $\Delta_i = f_i - f_1 > 0$ for some $i \in \{i \in \mathbb{N} \mid 2, \ldots, v\}$, because $f_1 < f_i$.

Note that $\|\mathbf{x}_1 - \mathbf{x}_i\| > 0$ $\forall i = 2, \ldots, v$, because we assume that the current simplex $S^{[q]}$ is an entire simplex.

As a result of Equation (6.12) and the fact that each $(\mathbf{x}_1 - \mathbf{x}_i)$ is a non-zero descent vector if $f_1 < f_i$, the vectorial sum $\sum_{i=2}^{v} (f_i - f_1) \frac{(\mathbf{x}_1 - \mathbf{x}_i)}{\|\mathbf{x}_1 - \mathbf{x}_i\|}$ must be composed of the sum of descent vectors, so the direction $\mathbf{d}^{[q]}$ given by Equation (6.9) is a descent vector of the objective function. ∎

A new reflection and expansion operation

As a consequence of a new reflection and expansion operation at any qth iteration, we have that the reflection trial point $\mathbf{x}_{ref}$ or the expansion trial point $\mathbf{x}_{\text{exp}}$ can be defined by

$$\mathbf{x}_\theta = \mathbf{x}_{from} + \theta l^{[q]} \mathbf{d}^{[q]}, \tag{6.13}$$

where $\mathbf{x}_{from}$ can be either $\mathbf{x}_{cen}$ or $\mathbf{x}_1$, $\mathbf{d}^{[q]}$ is the qth descent direction given by either Equation (6.8) or (6.9), depending on the chosen criterion, $l^{[q]} = \left\|\mathbf{x}_{cen}^{[q]} - \mathbf{x}_v^{[q]}\right\| > 0$ is the distance between $\mathbf{x}_{cen}^{[q]}$ and $\mathbf{x}_v^{[q]}$, and θ is given by

$$\theta = \begin{cases} \alpha & \text{Reflection operation} \\ \alpha\gamma & \text{Expansion operation} \end{cases}$$

Notice that the parameter α and γ are the reflection and expansion parameters of the conventional NM algorithm.

6.3.2 Analysis of the directional Nelder-Mead method

Without loss of generality, we shall study one iteration of the directional NM algorithm for a case of two dimensions in order to explain the behaviour of the method through this particular case. Suppose that the directional NM algorithm based on Equation (6.8) is applied to the problem of minimizing a non-linear objective function $f(\mathbf{x}) : \mathbb{R}^2 \rightarrow \mathbb{R}$, and at qth iteration we obtain the following entire simplex $S^{[q]}$

$$S^{[q]} = [\mathbf{x}_1 : \mathbf{x}_2 : \mathbf{x}_3] = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix}, \text{ where } f_1 \leq f_2 \leq f_3$$

For the current simplex, we have

$$l^{[q]} = \left\| S^{[q]} \left[\frac{1}{2}, \frac{1}{2}, -1\right]^T \right\|$$

For this particular case, $l^{[q]}$ is

$$l^{[q]} = \frac{\sqrt{(a_1+a_2)^2 + (b_1+b_2)^2 + 4a_3^2 + 4b_3^2 - 4(a_1+a_2)a_3 - 4(b_1+b_2)b_3}}{4}$$

The matrix $X^{[q]}$ and vector $\mathbf{f}^{[q]}$ are given by

$$X^{[q]} = \begin{bmatrix} \mathbf{1}_3 : & S^{[q]^T} \end{bmatrix} = \begin{bmatrix} 1 & a_1 & b_1 \\ 1 & a_2 & b_2 \\ 1 & a_3 & b_3 \end{bmatrix}, \ \mathbf{f}^{[q]} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}$$

Using Equation (6.11), we obtain

$$\beta^{[q]} = \frac{1}{\det(X^{[q]})} \begin{bmatrix} (a_2b_3 - a_3b_2)f_1 + (b_1a_3 - a_1b_3)f_2 + (a_1b_2 - a_2b_1)f_3 \\ (b_2 - b_3)f_1 + (b_3 - b_1)f_2 + (b_1 - b_2)f_3 \\ (a_3 - a_2)f_1 + (a_1 - a_3)f_2 + (a_2 - a_1)f_3 \end{bmatrix}$$

Note that $(a_2b_3 - a_3b_2)$ is the (1,1)-cofactor of $X^{[q]}$, $(b_1a_3 - a_1b_3)$ is the (2,1)-cofactor of $X^{[q]}$, and $(a_1b_2 - a_2b_1)$ is the (3,1)-cofactor of $X^{[q]}$.

By Equation (6.8), we obtain the normalized vector $\mathbf{d}^{[q]}$, whose coordinates are

$$\mathbf{d}^{[q]} = \begin{bmatrix} -g_1 \\ -g_2 \end{bmatrix} = \frac{-1}{\left\|[\beta_1, \beta_2]^T\right\|} \frac{1}{\det(X^{[q]})} \begin{bmatrix} (b_2 - b_3) f_1 + (b_3 - b_1) f_2 + (b_1 - b_2) f_3 \\ (a_3 - a_2) f_1 + (a_1 - a_3) f_2 + (a_2 - a_1) f_3 \end{bmatrix}. \tag{6.14}$$

If an expansion operation defined by Equation (6.13) occurs next, then the next $(q+1)$th simplex is given by

$$S^{[q+1]} = \begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & 1 & 0 \\ \frac{1}{2} & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} + \theta k_q \begin{bmatrix} -g_1 & 0 & 0 \\ -g_2 & 0 & 0 \end{bmatrix}$$

$$S^{[q+1]} = \begin{bmatrix} \frac{1}{2}a_1 + \frac{1}{2}a_2 - \theta g_1 l^{[q]} & a_1 & a_2 \\ \frac{1}{2}b_1 + \frac{1}{2}b_2 - \theta g_2 l^{[q]} & b_1 & b_2 \end{bmatrix}, \text{ where } \theta = \alpha\gamma \tag{6.15}$$

Furthermore, if the simplex $S^{[q]}$ were transformed by the classical NM method, we would obtain

$$S^{[q+1]} = \begin{bmatrix} \frac{1}{2}a_1(\theta + 1) + \frac{1}{2}a_2(\theta + 1) - \theta a_3 & a_1 & a_2 \\ \frac{1}{2}b_1(\theta + 1) + \frac{1}{2}b_1(\theta + 1) - \theta b_3 & b_1 & b_2 \end{bmatrix} \tag{6.16}$$

Observe that the $S^{[q+1]}$ given by Equation (6.15) depends on, among other parameters, f_1, f_2 and f_3, which could be vulnerable to changing of the objective function, especially when the objective function is corrupted by noise. In contrast, the $(q+1)$th simplex given by Equation (6.16) does not depend on the values of the objective function. However, the NM method makes decisions of the transformation of the simplex according to the value of the trial point that is obtained by the operations of the algorithm.

In Table 6.3 the directional NM method based on $\mathbf{x}_{from}$=$\mathbf{x}_{cen}$ is denoted by d-NM($\mathbf{x}_{cen}$) and the directional NM method based on $\mathbf{x}_{from}$=$\mathbf{x}_1$ is denoted by d-NM($\mathbf{x}_1$), where the descent direction is calculated by Equation (6.8). The parameters for all methods were fixed as follows: $\alpha = 0.95$; $\beta = 0.5$; $\gamma = 2$; $\delta = 0.5$ and $\tau = 1$. The tolerance error for stopping the algorithms were fixed with $\eta_1 = \eta_2 = 10^{-6}$, and the initial point for all numerical text was $\mathbf{x}_{ini} = 20\ \mathbf{1}_d$.

Table 6.3: Comparison of the NM method vs the directional NM methods

	NM		d-NM($\mathbf{x}_{cen}$)		d-NM($\mathbf{x}_1$)	
Objective function	NE	DTP	NE	DTP	NE	DTP
$\sum_{i=1}^{2} x_i^2$	2435	1.9E-150	1428	9.7E-90	2398	E-162
$\sum_{i=1}^{4} x_i^2$	3235	7.7E-63	745	1.5E-21	5582	E-162
2d-Rosenbrock	396	1.4E-6	1263	39.003	134	47.412
4d-Rosenbrock	2774	6E-6	747	40.316	5383	35.149

As can be seen from Table 6.3, the directional NM methods have a suitable behaviour in the case of the convex quadratic objective function. However, for the case of the Rosenbrock function in 2 and 4 dimensions, the methods do not work appropriately. That confirms the fact that the directional NM method is sensitive to changing of the objective function, as was previously explained by Equation (6.15). Hence the directional NM methods have better performance than the NM method when the objective function is convex, whilst for the case of the Rosenbrock function, the directional NM methods do not work correctly. However, further study of this approach can help us to explain other reasons for why the directional NM methods do not work correctly.

6.4 Conclusions

From comparative analysis between the two explored variations, we appreciate that the dynamic LCNM version has better performance than the directional NM version. However, the dynamic LCNM algorithm has not evidenced significant advantages over our original LCNM algorithm.

Despite the dynamic version of the LCNM algorithm requiring of the order of two function evaluations per iteration, the dynamic $\hat{\gamma}$-LCNM algorithm appeared to display an adequate performance in comparison with the original LCNM algorithm.

However, a comparative study of the algorithms here developed, in order to find the best settings of the algorithms is nowadays an open research problem.

Chapter 7

Constrained optimization of noisy functions

7.1 Introduction

A comparative study of the LCNM method and the LCNM+PC method is presented in this chapter, when the objective function is in the presence of error. This kind of problem occurs in optimization problems of simulation of stochastic systems, aimed at the study of the improvement of real complex system, such as, manufacturing systems, traffic light systems and PERT-project networks.

This chapter is concerned with a comparative analysis between the LCNM method and the LCNM+PC method from the standpoint of their performance, when they are applied to noisy objective function subject to linear constraints. For this, a brief review of non-parametric statistical comparison methods and a modified test developed by Brown and Forsythe (1974) for testing the equality of variances are presented in Section 7.2. We describe in Section 7.3 the employed test problems for contrasting the LCNM method and the LCNM+PC method, the statistical test for equality of variances and the results of the comparisons by differences of percentiles, median and mean. Plots of cumulative frequencies of the DTP's and of the PM for both methods are shown in section 7.4, enriching the comparative analysis. Finally, Section 7.5 gives a discussion of the comparative analysis and conclusions of the study of the developed methods.

7.2 Review of statistical comparison methods

We shall herein include a basic review of the methods of comparison from the statistical analysis viewpoint, because the LCNM and the LCNM+PC methods will be compared using a set of test problems, where each objective function contains a noisy function given by a normal distribution with mean zero and variance σ^2. Though only normal errors have been used, both versions of the method have been designed for identifying the optimum, even if the noisy function is characterized by other distributions.

Since the methods were applied to linearly constrained noisy optimization problems, it was necessary to make a number of independent replications or samples for obtaining a statistical analysis of their performance and accuracy. Parametric statistical comparison methods of means usually suppose that the error distribution associated with each sample must be independent and the response follows a known distribution (Hsu 1996). In our case, this latter assumption may not be satisfied.

Hence, we use a non-parametric statistical method (sometimes called a distribution-free method).

In particular, the distribution-free rank sum (DFRS) test (the two-sample Wilcoxon rank sum test) allows one to investigate the presence of treatment effect, as a result of a shift of location between two data groups. An important feature of the DFRS test is its applicability to unpaired samples (Mann and Whitney 1974), as is the case for data from our experiments. However, it must satisfy the following assumptions:

- The m observations $X_1, \ldots X_m$ are an independent and identically distributed random sample from population 1, say the control population. The n observations $Y_1, \ldots Y_n$ also are an independent and identically distributed random sample from population 2, called the treatment population.
- Both populations, control and treatment, are continuous.

The DFRS test establishes as null hypothesis that both distribution functions, control distribution $F(t)$ and treatment distribution $G(t)$, have the same probability distribution. On the other hand, its alternative hypothesis asserts that the distribution function of the treatment population $G(t)$, is shifted with respect to the distribution function of the treatment population $F(t)$, by the amount Δ.

Furthermore, if the DFRS test assumes that $F(t) = G(t - \Delta)$ for all t, then the independent random sample data from two populations must be of the same shape approximately. Therefore, the variances of both populations must be approximately equal.

Namely,

$$\mathrm{H}_0\colon\ F(t) = G(t) \ \text{ vs } \ \mathrm{H}_1\colon\ F(t) = G(t - \Delta) \quad \forall t \tag{7.1}$$

The alternative hypothesis H_1 written in Equation (7.1) says control and treatment populations are the same except by the amount Δ, whose value represents the effect due to the treatment. This is, if $\Delta > 0$, then the expected value of the treatment population is more than the expected value of the control population. Whereas, if $\Delta < 0$, then the expected value of the treatment population decrease due to the treatment. In mathematical terms, if $E(X)$ and $E(Y)$ are the means of control and treatment populations, respectively, then

$$\Delta = E(X) - E(Y) \tag{7.2}$$

Equation (7.2) is a consequence of writing this as $Y \stackrel{d}{=} X + \Delta$, where the symbol $\stackrel{d}{=}$ represents "the same distribution", and X and Y are the random variables corresponding to each population.

In the comparison of the performance of the LCNM method whose observations were based on the control population versus, the performance of the LCNM+PC methods for establishing the treatment effect, we applied the statistical test of an one-sided upper-tail for large number of observations, using the Wilcoxon two-sample rank sum statistic W^* given by

$$W^* = \frac{W - E_0[W]}{(Var_0[W])^{1/2}},$$

where W is the Wilcoxon two-sample rank statistics, and $E_0[W]$ and $Var_0[W]$ corresponding to the mean and the variance of W respectively, when the null hypothesis is true. These are given by

$$E_0[W] = \frac{n(m+n+1)}{2} \qquad Var_0[W] = \frac{mn(m+n+1)}{12}$$

The hypotheses of one-sided upper-tail test are defined as:

$$\mathrm{H}_0\colon\ \Delta = 0 \text{ vs } \mathrm{H}_1\colon\ \Delta > 0,$$

where for a level of significance $\tilde{\alpha}$, H_0 is rejected if $W^* \geq z_{\tilde{\alpha}}$, otherwise H_0 cannot be rejected H_0. Here $z_{\tilde{\alpha}}$ is the critical value of a normal distribution $N(0,1)$ (Hollander and

Wolfe 1999).

If the populations have different standard deviations, a two-sample t-test without pooling variances may be more appropriate (Ryan et al. 1985) for comparing the sample means $\bar{x}_1$ and $\bar{x}_2$ of each population using the statistic t, given by

$$t = \frac{\bar{x}_1 - \bar{x}_2}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}},$$

and its degrees of freedom, $d.f.$, is based on the following approximation,

$$d.f. = \frac{\left[(s_1^2/n_1) + (s_2^2/n_2)\right]^2}{\frac{(s_1^2/n_1)^2}{n_1-1} + \frac{(s_2^2/n_2)^2}{n_2-1}},$$

where $\bar{x}_i$, s_i^2 and n_i are the sample mean, sample variance and sample size, respectively, of the ith population.

In our case, because the samples yielded from each experiment have the same number of observations n, the two-sample t-test approach is acceptably safe, even if the variances differ (Law and Kelton 2000).

Since our interest is centred on the use of the statistical test for comparing the populations that emanate from the results of the test problems, we shall not consider theoretical aspects of the test in this work. A detailed explanation of the employed test procedure is widely presented in (Hollander and Wolfe 1999) and (Arnold 1990).

To do the statistical tests, we employed the well-known statistical software Minitab® (Minitab 1994), which offers a flexible interface with some spreadsheet software such as Microsoft® Excel. The DFRS test can be performed using Minitab, estimating the equality of two population medians, and calculating the corresponding point estimate and its confidence interval.

The hypotheses that can be tested through Minitab are H_0: $\mu_1 = \mu_2$ versus H_1: $\mu_1 \neq \mu_2$ ($\mu_1 < \mu_2$ or $\mu_1 > \mu_2$) for two-sided test (one-sided lower-tail test or one-sided upper-tail test), where μ is the population median, which is also denoted by $md[x]$. In the case that there exist ties in the data, Minitab adjusts the significance level (Ryan et al. 1985).

Minitab can also carry out the two-sample t-test under the assumption of equal or unequal variances, which may be considered as a way for comparing populations when their variances are different (Ryan et al. 1985). This latter comparison approach was also performed for obtaining more information, when the variances of the populations to be compared are statistically distinct.

In addition Minitab allows one to perform the F-test for comparing the variance of two normal populations, and the test of Levene for equality of variances of two either normal or non-normal populations using the modified procedure of Levene developed by Brown and Forsythe (1974). This latter method regards the distances of the observations from their sample median rather than their sample mean, making procedure more robustly asymptotically distribution-free.

The hypotheses that can be tested using Minitab for the F-test and the test of Levene are H_0: $\sigma_1^2 = \sigma_2^2$ versus H_1: $\sigma_1^2 \neq \sigma_2^2$, where σ_1^2 and σ_2^2 are the variances of the population 1 and population 2, respectively. If the reported p-value of Minitab is less than the level of significance chosen by the user, then the null hypothesis of equal variance can be rejected (Minitab 2000).

In our case, we shall compare two groups of data $x_{ij} = \mu_i + \epsilon_{ij}$ $(i = 1, 2)$ of the same sample size N, where μ_i is unknown and μ_1 and μ_2 are assumed unequal, and ϵ_{ij} are independent and the same distribution with means zero and not necessarily equal variances. The Levene test statistic is given by

$$L = \frac{(N-k)\sum_{i=1}^{2} N(\bar{V}_{i.} - \bar{V}_{i..})^2}{(k-1)\sum_{i=1}^{2}\sum_{j=1}^{N}(V_{ij} - \bar{V}_{i.})^2},$$

where $k = 2$ is the number of groups, V_{ij} is estimated by Minitab using the following definition

$$V_{ij} = |x_{ij} - \tilde{x}_{i.}| \quad \text{where } \tilde{x}_{i.} \text{ is the median of each } i\text{th group},$$

$\bar{V}_{i.}$ is the ith mean of the V_{ij}, and $\bar{V}_{i..}$ is the overall mean of all sample V_{ij} (Brown and Forsythe 1974). Moreover, the estimation of V_{ij} using the median provides enough robustness when the data are non-normal distribution.

7.3 Comparison of the LCNM versus the LCNM+PC methods

To compare the effectiveness and accuracy of these algorithms, we created a set of noisy objective functions or artificial response functions, whose value depends on: a deterministic function say $f(\mathbf{x})$, and a stochastic term denoted as $\mathcal{N}(\sigma)$. This latter was generated through a normal distribution pseudo-random number generator with mean zero and variance σ^2. Equation (7.3) represents a general response function employed in the computational experiments.

$$\tilde{f}(\mathrm{x}) = f(\mathrm{x}) + \mathcal{N}(\sigma) \tag{7.3}$$

Two main computational experiments were performed to determine the properties of the LCNM and the LCNM+PC methods, when the cone angle of the feasible region is either sharp or obtuse, and when the best local minimum is located inside feasible region or on the boundary. The level of noise was controlled by the value of σ.

7.3.1 Description of the tests of comparison

To contrast both methods, a set of test problems was carried out, which is grouped by: the type of region, location of the best local optimum, deterministic objective function and noise level. This taxonomical scheme is depicted in Figure 7.1.

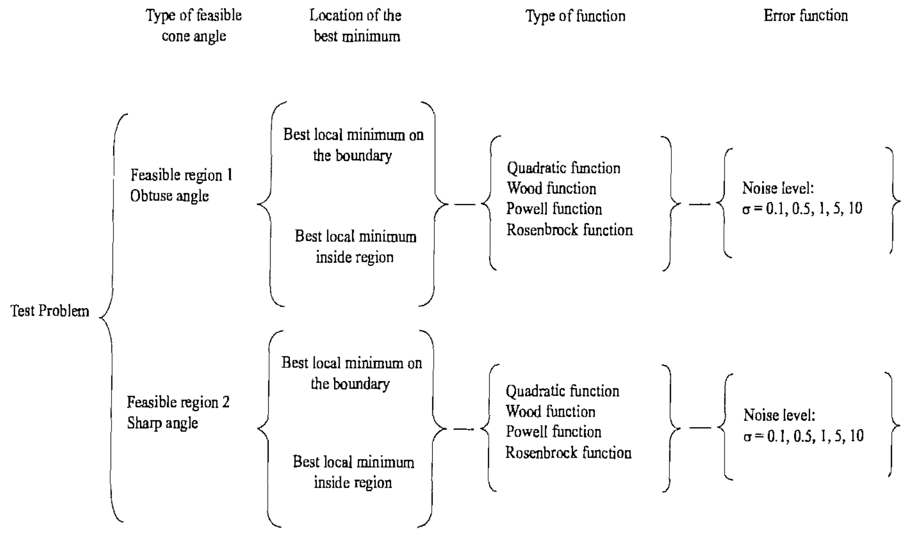


Figure 7.1: Taxonomy of test problems for noisy function cases.

To compare the computational effort of both methods, it we used the performance measure given by

$$PM_i = NE_i + 100 \cdot DTP_i \quad \forall i = 1, 2, 3, 4, 5$$

where NE_i is the number of function evaluations carried out during the search of the minimum, DTP_i is the distance to the true point for each test and the subscript ith represents the fixed noise level. Observe that the best algorithm is that whose PM is the smallest.

The accuracy of both methods was measured through the sampling of the DTP for each noise level for comparing both methods.

Table 7.1 displays the blocks of 100 independent observations by symbol ✓ for each test problem grouped by obtuse and sharp cone angle of the feasible region, by the noise level. This experimental structure was also used for obtaining DTP and NE data. The notation of the test problems indicated in the tables is as defined in Subsection 5.4.1.

Table 7.1: Blocks of experiments for the feasible region 1 and region 2.

			PM_1	PM_2	PM_3	PM_4	PM_5
	Setting	Test \ σ	0.1	0.5	1	5	10
R	$\alpha = 0.95$, $\tau = 0.5$	TP1(4)	✓	✓	✓	✓	✓
e	$\alpha = 0.95$, $\tau = 0.5$	TP7(4)	✓	✓	✓	✓	✓
g	$\alpha = 0.95$, $\tau = 1$	TP9(4)	✓	✓	✓	✓	✓
i	$\alpha = 0.95$, $\tau = 1$	TP5(2)	✓	✓	✓	✓	✓
o	$\alpha = 0.95$, $\tau = 0.5$	TP2(4)	✓	✓	✓	✓	✓
n	$\alpha = 0.95$, $\tau = 0.5$	TP8(4)	✓	✓	✓	✓	✓
	$\alpha = 0.95$, $\tau = 1$	TP10(4)	✓	✓	✓	✓	✓
1	$\alpha = 0.95$, $\tau = 1$	TP6(2)	✓	✓	✓	✓	✓
R	$\alpha = 0.95$, $\tau = 0.5$	TP11(4)	✓	✓	✓	✓	✓
e	$\alpha = 0.95$, $\tau = 0.5$	TP17(4)	✓	✓	✓	✓	✓
g	$\alpha = 0.95$, $\tau = 0.5$	TP19(4)	✓	✓	✓	✓	✓
i	$\alpha = 0.95$, $\tau = 0.5$	TP15(2)	✓	✓	✓	✓	✓
o	$\alpha = 0.95$, $\tau = 0.5$	TP12(4)	✓	✓	✓	✓	✓
n	$\alpha = 0.95$, $\tau = 0.5$	TP18(4)	✓	✓	✓	✓	✓
	$\alpha = 0.95$, $\tau = 0.5$	TP20(4)	✓	✓	✓	✓	✓
2	$\alpha = 0.95$, $\tau = 0.5$	TP16(2)	✓	✓	✓	✓	✓

The groups of test problems were carried out using the LCNM and the LCNM+PC methods under the same algorithm settings with stopping parameters $\eta = 0.1$, $\eta_1 = \eta_2 = 10^{-6}$ and $\Delta = 10^{-5}$. The values of α (reflection coefficient of the LCNM) and τ (parameter of step size) are also shown in the above tables. As initial point we used the point $20 \cdot \mathbf{1}_d$ for all tests.

A scheme of classification of the test problems is given in the following table. Note that they are grouped by feasible region class and location of the assumed best local minimum.

Table 7.2: Taxonomy of the test problems.

	True point on the boundary	True point inside the region
Region 1: Obtuse angle	TP1(4), TP7(4) TP9(4), TP5(2)	TP2(4), TP8(4) TP10(4), TP6(2)
Region 2: Sharp angle	TP11(4), TP17(4) TP19(4), TP15(2)	TP12(4), TP18(4) TP20(4), TP16(2)

7.3.2 Statistical test for equality of variances

To verify the assumption of equality of distribution shape that must be satisfied by the populations for using the DFRS test, we carried out the comparison variance test of Minitab, which allows us to obtain a report of both the F-test under the assumption of a normal distribution and the test of Levene for any continuous distribution. This latter is a modification developed by Brown and Forsythe (1974).

Since in most cases the populations are non-normal, it was appropriate to report the test of Levene with 95% confidence interval, under the assumption of similarity of the distributions. Thus, if the p-value of each test estimated by Minitab is less than 0.05, then we rejected the null hypothesis of equality variances.

Table 7.3: Test of Levene for equal variances of the PM's in the feasible region 1.

	Method	PM_1	PM_2	PM_3	PM_4	PM_5
TP1(4)	L statistic	6.016	2.430	0.414	2.115	4.620
	p-value	0.015	0.121	0.521	0.147	0.033
TP7(4)	L statistic	11.908	2.558	2.337	1.744	3.018
	p-value	0.001	0.111	0.128	0.188	0.084
TP9(4)	L statistic	30.704	8.938	0.025	0.441	0.056
	p-value	0.000	0.003	0.875	0.507	0.813
TP5(2)	L statistic	0.065	0.051	1.440	1.519	2.916
	p-value	0.799	0.821	.0232	0.219	0.089
TP2(4)	L statistic	37.680	26.370	25.719	26.074	4.727
	p-value	0.000	0.000	0.000	0.000	0.031
TP8(4)	L statistic	72.520	6.577	14.142	18.220	72.522
	p-value	0.000	0.011	0.000	0.000	0.000
TP10(4)	L statistic	8.677	17.838	230.427	21.672	4.036
	p-value	0.004	0.000	0.000	0.000	0.046
TP6(2)	L statistic	0.682	2.461	1.069	0.838	3.493
	p-value	0.410	0.118	0.302	0.361	0.063

Test statistics and p-values of the test of Levene are shown in Table 7.3, when the feasible region is the an obtuse angle, for comparing the variance of each PM_i distribution of the LCNM and the LCNM+PC methods for each test problem. As can be seen from the table, most of the p-values are less than 0.05, so in these cases the null hypothesis that there is the equality of variances of the populations is rejected.

Table 7.4: Test of Levene for equal variances of the PM's in the feasible region 2.

	Method	PM_1	PM_2	PM_3	PM_4	PM_5
TP11(4)	L statistic	26.034	8.709	1.974	2.907	1.873
	p-value	0.000	0.004	0.162	0.090	0.173
TP17(4)	L statistic	20.593	59.183	86.507	28.481	45.564
	p-value	0.000	0.000	0.000	0.000	0.000
TP19(4)	L statistic	42.333	1.439	0.979	8.800	45.087
	p-value	0.000	0.232	0.324	0.003	0.000
TP15(2)	L statistic	118.612	81.167	165.910	85.010	218.248
	p-value	0.000	0.000	0.000	0.000	0.000
TP12(4)	L statistic	5.471	0.231	1.334	6.307	0.690
	p-value	0.020	0.631	0.250	0.013	0.407
TP18(4)	L statistic	5.580	79.236	108.859	17.112	14.960
	p-value	0.019	0.000	0.000	0.000	0.000
TP20(4)	L statistic	46.157	31.686	1.592	3.262	0.535
	p-value	0.000	0.000	0.208	0.072	0.465
TP16(2)	L statistic	6.537	12.016	17.279	21.293	4.340
	p-value	0.011	0.001	0.000	0.000	0.039

Table 7.4 displays the test statistics and the p-values of the test of Levene for the set of test problems when the feasible region has a sharp solid angle. According to this table, most of the p-values are less than 0.05. Therefore, in these cases, we reject the null hypothesis of equality of variances of the populations.

Even where the null hypothesis of equality of variances of the populations cannot be rejected, it does not mean that the populations are of the same shape approximately.

7.3.3 Comparison by 90th percentile

Since the population of PM_i's are non-normal distributions and in most cases the distribution of the $PM_{i,\text{LCNM}}$ and $PM_{i,\text{LCNM+PC}}$ are of unequal variances for each ith noise level, we therefore compared the performance of both algorithms by 90th percentile of each PM_i distribution. Tables 7.5 and 7.6 report the percentage relative changing $\delta P_i = 100 \cdot (P_{i,\text{LCNM}} - P_{i,\text{LCNM+PC}})/P_{i,\text{LCNM}}$ for each level noise $i = 1, ..., 5$, where $P_{i,\text{LCNM}}$ is the 90th percentile of the PM_i when we applied the LCNM algorithm to each test problem and, $P_{i,\text{LCNM+PC}}$ is the 90th percentile of the PM_i when we applied the LCNM+PC algorithm.

Table 7.5: Percentage relative changing for the 90th percentile of the PM's in region 1.

	δP_1	δP_2	δP_3	δP_4	δP_5
TP1(4)	5.84	12.02	9.75	3.63	-1.12
TP7(4)	0.84	7.47	3.86	-10.07	-13.37
TP9(4)	10.13	10.30	9.05	0.14	1.08
TP5(2)	7.14	30.63	1.80	-1.70	-4.15
TP2(4)	22.25	22.47	22.53	15.15	6.70
TP8(4)	-9.15	-5.67	-11.02	85.52	86.61
TP10(4)	-22.33	33.80	35.21	14.58	7.97
TP6(2)	7.90	30.13	11.67	-1.17	-1.85

Table 7.6: Percentage relative changing for the 90th percentile of the PM's in region 2.

	δP_1	δP_2	δP_3	δP_4	δP_5
TP11(4)	34.58	31.18	25.03	18.29	5.06
TP17(4)	29.31	73.04	73.60	69.21	59.27
TP19(4)	43.58	39.26	38.12	57.78	94.19
TP15(2)	60.71	61.40	61.40	61.34	60.71
TP12(4)	-18.57	-12.02	12.89	6.92	1.40
TP18(4)	46.39	68.63	68.20	40.43	13.02
TP20(4)	-55.98	-51.72	-45.43	-10.25	-0.84
TP16(2)	40.57	27.41	25.28	26.33	26.14

From Tables 7.5 and 7.6, we observe that in 77% of the cases the values of δP_i are positive. So the performance measure of the LCNM algorithm was greater than the performance measure of the LCNM+PC algorithm in most cases. This is evidence that the LCNM+PC algorithm has advantages over its original version, when the objective function is altered by noise in 4-dimension cases. Obviously, this comparative study cannot be regarded as a definite proof of the superiority of the LCNM+PC algorithm. However, it can help us to establish some comparative features.

7.3.4 Comparison by median and mean

Despite the fact that the populations of PM_i can have different shapes and variances in most of the cases, we supposed true the assumptions of the DFRS test for performing the non-parametric statistical test. Table 7.7 and Table 7.8 show the 95% confidence intervals for the difference of the median of PM_i of the LCNM and the LCNM+PC methods. Thus $md[\Delta PM_i] = md[PM_{i,\text{LCNM}} - PM_{i,\text{LCNM+PC}}]$ for $i = 1, ..., 5$ are reported for the DFRS test, where $md[\cdot]$ represents the median.

Table 7.7: Confidence intervals for difference of the median of the PM's in the region 1.

	$md[\Delta PM_1]$	$md[\Delta PM_2]$	$md[\Delta PM_3]$	$md[\Delta PM_4]$	$md[\Delta PM_5]$
TP1(4)	(9.16, 37.23)	(10.05, 39.00)	(12.35, 44.68)	(-22.51, 30.84)	(-45.17, 25.11)
TP7(4)	(-15.66, -9.96)	(25.94, 33.97)	(21.01, 30.32)	(-3.38, 24.43)	(-36.71, -14.14)
TP9(4)	(16.96, 22.58)	(21.66, 26.68)	(21.39, 26.85)	(4.45, 23.91)	(0.47, 24.60)
TP5(2)	(11.11, 19.47)	(38.0, 691.3)	(-81.3, 508.4)	(-145.5, 39.6)	(-74.0, 64.9)
TP2(4)	(-54.12, -26.17)	(-66.22, -39.48)	(-50.14, -26.45)	(-12.74, 27.82)	(-12.01, 33.07)
TP8(4)	(103.15, 139.59)	(135.26, 149.01)	(-22.61, 98.66)	(-45.8, 22.4)	(630.1, 4135.6)
TP10(4)	(-56.19, -49.26)	(-56.67, -44.93)	(19.40, 114.30)	(-2.10, 49.50)	(-3.86, 37.44)
TP6(2)	(-9.39, 10.94)	(459.1, 1042.9)	(116.1, 669.0)	(-124.3, 53.4)	(-123.3, 32.3)

Table 7.8: Confidence intervals for difference of the median of the PM's in the region 2.

	$md[\Delta PM_1]$	$md[\Delta PM_2]$	$md[\Delta PM_3]$	$md[\Delta PM_4]$	$md[\Delta PM_5]$
TP11(4)	(17.05, 58.91)	(49.48, 87.66)	(50.38, 88.77)	(47.62, 89.83)	(23.64, 76.51)
TP17(4)	(254.06, 291.53)	(299.5, 355.5)	(363.8, 1913.2)	(162.3, 221.8)	(75.9, 165.9)
TP19(4)	(180.21, 198.44)	(167.59, 178.76)	(144.60, 171.98)	(212.55, 160.18)	(132.7, 229.6)
TP15(2)	(31.00, 33.00)	(32.00, 33.00)	(31.00, 33.00)	(32.00, 33.00)	(28.00, 32.00)
TP12(4)	(-47.10, -30.07)	(-44.14, -19.21)	(-22.63, 11.22)	(-2.11, 40.94)	(-5.22, 38.24)
TP18(4)	(605.9, 639.5)	(678.5, 855.1)	(216.1, 519.8)	(35.85, 114.43)	(21.01, 88.67)
TP20(4)	(-151.66,-146.24)	(-137.28,-131.38)	(-129.00,-122.41)	(-103.55,-81.46)	(-38.78, 7.34)
TP16(2)	(116.46, 135.27)	(83.71, 100.55)	(77.79, 96.30)	(93.98, 108.55)	(66.44, 84.56)

From Table 7.7, we observe that the LCNM algorithm has better performance than its modified version for obtuse angle cases, because the number of negative confidence intervals is 27 of a total of 40 experiments. However, when the solid angle of the feasible region is sharp, the LCNM+PC method displays superiority over its original algorithm, due to the fact shown in Table 7.8, that there exist 30 positive confidence intervals of 40 experiments. This means that the 75% of the cases, the PM of the LCNM+PC method was better (smaller) that the PM of the LCNM method.

In addition we carried out the two-sample t-test without pooling variances and with 95% lower bound confidence intervals for the difference of the median, at each noise level. In Table 7.9 and Table 7.10, we report 95% lower bound confidence intervals for the difference displayed in square brackets, and for the point estimate of the population difference mean $\Delta\hat{E}[PM_i] = \hat{E}[PM_{i,\mathrm{LCNM}}] - \hat{E}[PM_{i,\mathrm{LCNM+PC}}]$.

Note that in some cases, the populations are non-normal distributions, which do not satisfy the assumption of the two-sample t-test. However, the two-sample t-test without pooling variances may nevertheless be useful for comparing the performance of both methods, even when the population are non-normal distribution.

In Section 7.4 are depicted a set of cumulative frequencies of both DTP's and the PM's by test problem and noise level, when the LCNM and the LCNM+PC methods are applied. As may be seen from the cumulative frequencies, some PM's display a clearly non-normal distribution, such as the test problems TP8(4) and TP17(4) when the LCNM method was applied. An examination of the different cumulative frequencies of all the carried out test problems clearly verifies that the test problems TP8(4) and TP17(4) are from a non-normal distribution. The rest of them could be considered normally distributed.

Table 7.9: Summary of two-sample t-tests for 95% lower bound confidence intervals and the point estimates of the population difference mean of the PM's for the region 1.

	$\Delta\hat{E}[PM_1]$	$\Delta\hat{E}[PM_2]$	$\Delta\hat{E}[PM_3]$	$\Delta\hat{E}[PM_4]$	$\Delta\hat{E}[PM_5]$
TP1(4)	[6.33] 19.41	[15.92] 28.43	[15.17] 28.22	[-15.1] 13.7	[-40.3] -12.7
TP7(4)	[-12.47] -8.36	[22.00] 26.37	[17.56] 22.40	[-7.83] 0.66	[-34.83] -26.05
TP9(4)	[18.73] 20.81	[22.63] 24.59	[19.78] 23.50	[3.63] 13.17	[1.08] 11.45
TP5(2)	[-22.5] 7.0	[143] 384	[-31] 193	[-133.7] -39.0	[-90.6] -26.3
TP2(4)	[-24.1] -6.8	[-37.81] -22.02	[-30.50] -15.98	[-4.29] 12.10	[-9.1] 9.1
TP8(4)*	[44.4] 63.3	[78.89] 95.11	[4.5] 28.8	[375] 634	[2058] 2404
TP10(4)	[-55.58] -52.85	[-31.36] -15.68	[54.7] 73.0	[6.5] 24.2	[1.7] 19.8
TP6(2)	[-2.8] 19.9	[591] 846	[153] 368	[-114.4] -22.7	[-95.7] -20.9

From Table 7.9, the LCNM+PC method had better (smaller) performance than the LCNM method, in most cases when the best optimum point is on the boundary of the feasible region, especially in the cases with noise level of $\sigma = 0.5$, 1 and 5. On the other hand, for the cases when the best local optimum is inside the feasible region, there does not exist a clear evidence of the advantage of one of the methods. Nonetheless, when the noise level is high, the PM of the LCNM+PC method is smaller than the PM of the LCNM method.

Table 7.10: Summary of two-sample t-tests for 95% lower bound confidence intervals and the point estimates of the population difference mean of the PM's for the region 2.

	$\Delta\hat{E}[PM_1]$	$\Delta\hat{E}[PM_2]$	$\Delta\hat{E}[PM_3]$	$\Delta\hat{E}[PM_4]$	$\Delta\hat{E}[PM_5]$
TP11(4)	[33.5] 51.5	[59.0] 75.7	[54.29] 70.78	[51.9] 68.6	[27.7] 48.4
TP17(4)*	[213.6] 237.4	[615.7] 746.5	[983.0] 1130.6	[274.8] 376.1	[210.8] 299.9
TP19(4)	[180.84] 189.78	[157.50] 167.53	[145.11] 155.17	[260] 448	[1163] 1524
TP15(2)	[30.618] 31.200	[31.225] 31.810	[30.434] 31.080	[30.653] 31.24	[29.836] 30.44
TP12(4)	[-59.54] -47.47	[-39.78] -27.79	[-16.91] -3.66	[1.3] 18.9	[0.8] 21.0
TP18(4)	[630.9] 684.0	[707.7] 807.1	[451.4] 544.6	[85.3] 135.0	[43.3] 73.3
TP20(4)	[-150.86] -148.52	[-136.15] -133.60	[-126.14] -121.43	[-89.71] -76.22	[-35.8] -15.1
TP16(2)	[122.07] 128.17	[89.02] 96.54	[80.33] 88.54	[96.53] 104.33	[71.87] 79.48

As can be seen in Table 7.10, the LCNM+PC method had better performance than the LCNM method when the best local optimum is on the boundary of the feasible region, for all the noise levels. However, this advantage of the LCNM+PC method over the LCNM method cannot be affirmed in the other cases.

7.4 Contrasting the methods by cumulative frequency curves

We now compare both methods using by considering the curves of cumulative frequencies of the PM_i and the DTP_i for all noise levels. In particular, we concentrate on the test problem when the objective function is a convex quadratic function and the Wood function.

Table 7.11: Comparison of the test problems by cumulative frequency curvers.

	True point on the boundary	True point inside the region
Obtuse angle (Region 1)	TP1(4), TP7(4)	TP2(4), TP8(4)
Sharp angle (Region 2)	TP11(4), TP17(4)	TP12(4), TP18(4)

Table 7.11 displays the set of test problems that were employed for comparing the accuracy and efficiency in terms of the PM of the developed methods, when an error alters the objective function.

7.4.1 Case: Quadratic objective function

From Figure 7.2 and Figure 7.3, we observe how the LCNM+PC method is more accurate than the LCNM method, because the family of cumulative frequencies of the DTP when we applied the LCNM+PC method reach their maximum within an interval narrower than when we applied the LCNM method. Moreover, the PM of the LCNM is slightly smaller than the PM of the LCNM+PC.

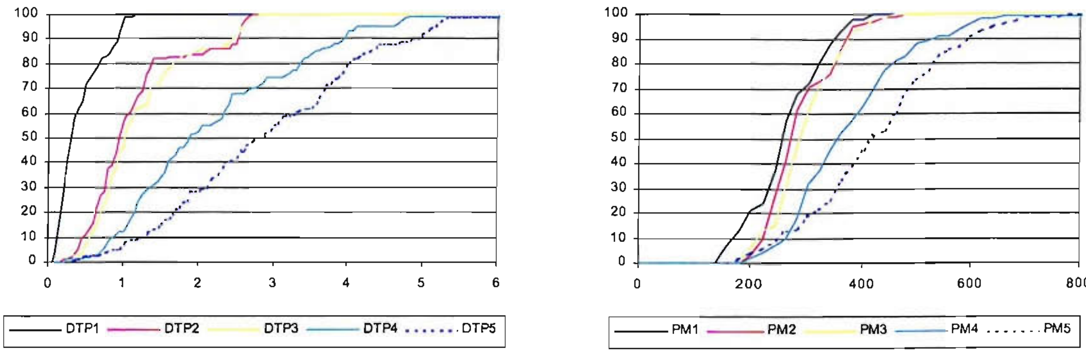


Figure 7.2: Curves of cumulative frequencies of the DTP and the PM for the TP1(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

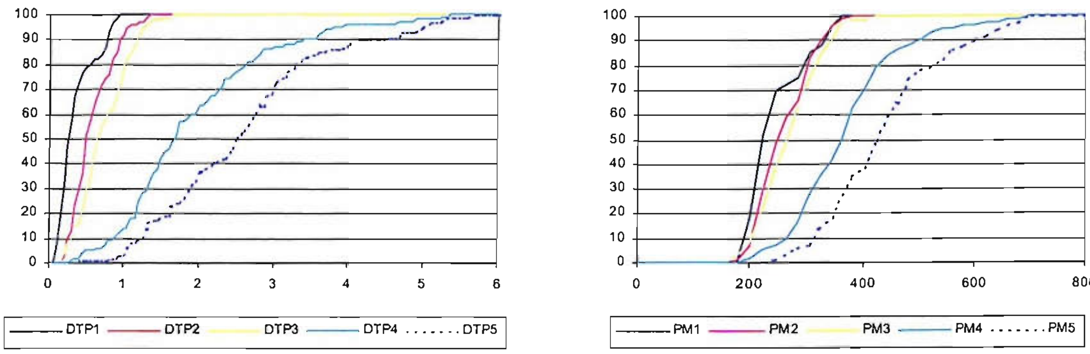


Figure 7.3: Curves of cumulative frequencies of the DTP and the PM for the TP1(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

As a result of a comparative analysis between Figure 7.4 and Figure 7.5, we note that the LCNM+PC method is more effective that its original version, because it converges to the true point, which is inside the feasible region, with more accuracy and better PM than the LCNM method. Furthermore, if the obtained figures of TP1(4) and TP2(4) are compared, we observe that the LCNM method had better performance in the case when the true point is on the boundary than when the true point is inside the feasible region. Nevertheless, the LCNM+PC method presented better performance when the true point is inside the feasible region than when the true point is on the boundary of the feasible region.

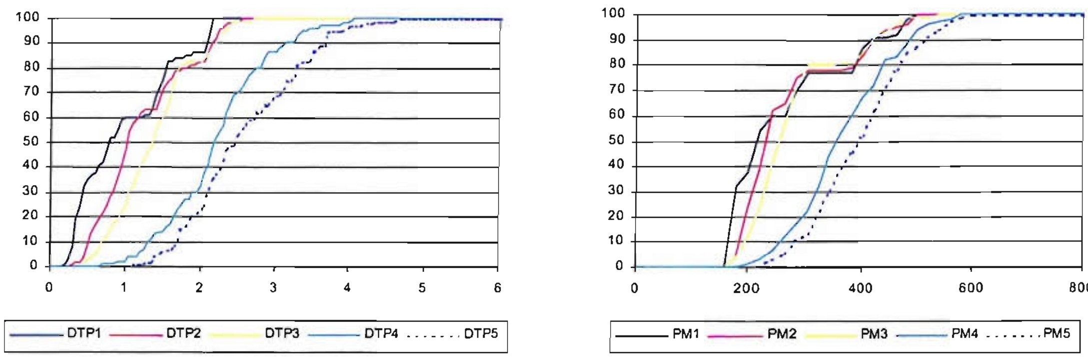


Figure 7.4: Curves of cumulative frequencies of the DTP and the PM for the TP2(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

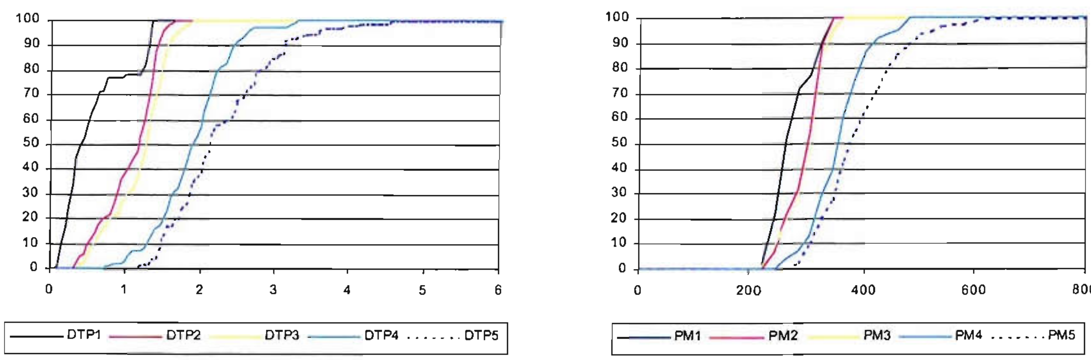


Figure 7.5: Curves of cumulative frequencies of the DTP and the PM for the TP2(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

The following figures correspond to the subgroup of test problems when the feasible region contains a sharp angle and the true point is on the boundary of the feasible region and inside the feasible region.

As are shown in Figure 7.6 and Figure 7.7, which correspond to the case when the true point is on the boundary, both the PM of the LCNM+PC method and its accuracy are better than when we applied the LCNM method.

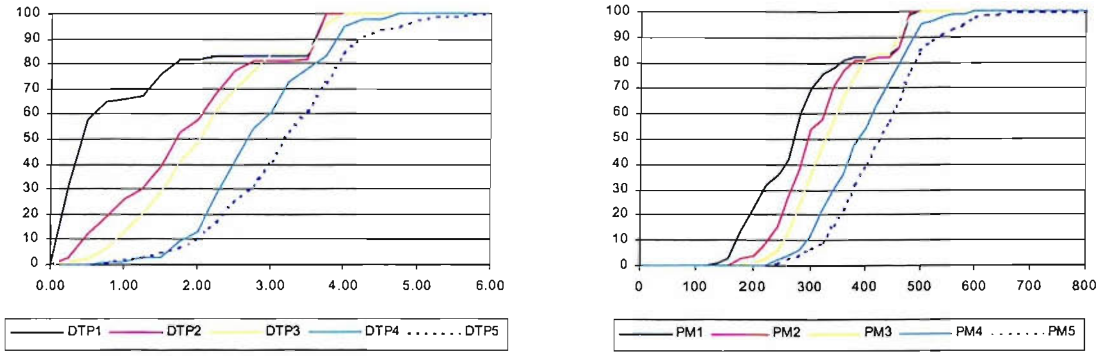


Figure 7.6: Curves of cumulative frequencies of the DTP and the PM for the TP11(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

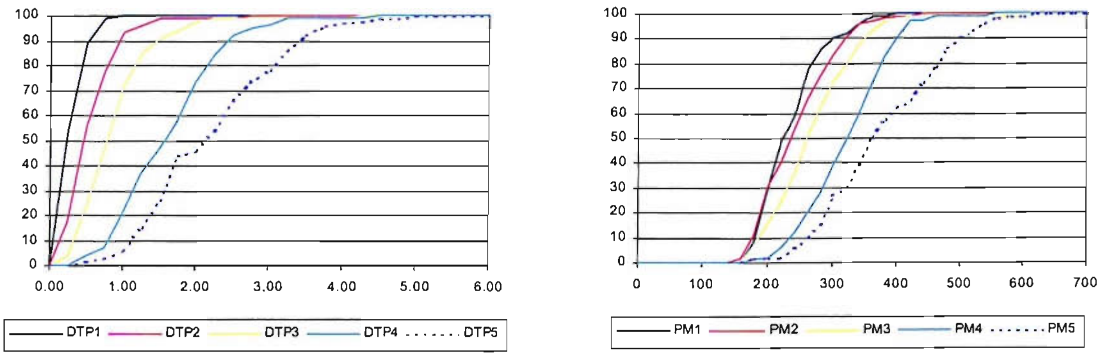


Figure 7.7: Curves of cumulative frequencies of the DTP and the PM for the TP11(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

It may be seen from Figure 7.8 and Figure 7.9 that the PM of the LCNM method is smaller than the PM of the LCNM+PC method. However, the LCNM+PC method is more accurate than the LCNM method. Note that the LCNM method reported better performance when the true point is inside the feasible region than when it is on the boundary. However, the PM of the LCNM+PC method is slightly less in the case when the true point is located on the boundary than when it is inside the feasible region.

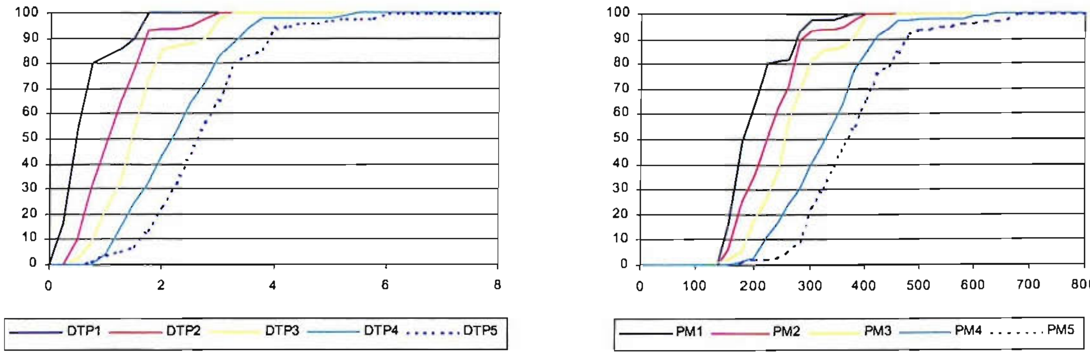


Figure 7.8: Curves of cumulative frequencies of the DTP and the PM for the TP12(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

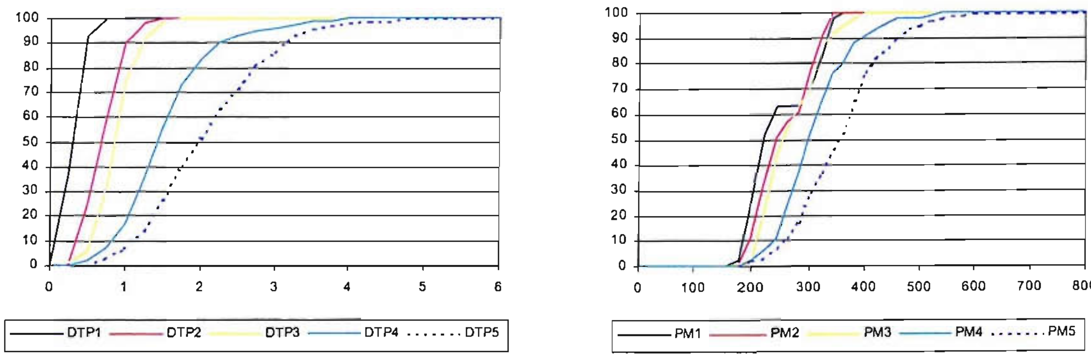


Figure 7.9: Curves of cumulative frequencies of the DTP and the PM for the TP12(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

7.4.2 Case: Wood objective function

As may be seen from Figure 7.10 and Figure 7.11 the LCNM+PC method has better performance from the point of view of its accuracy, although its PM is larger than the PM of the LCNM. Note that the cumulative frequency of the DTP of Figure 7.10 shows that the LCNM method converges in an important number of cases to another local minimum when the level of noise are 0.5, 1 and 5. In contrast, the LCNM+PC method converges to point close to the true point, even when the levels of noise are 0.5 and 1.

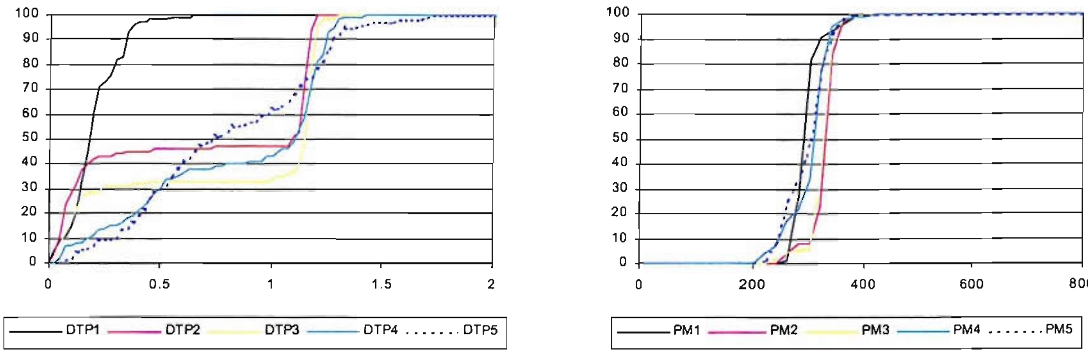


Figure 7.10: Curves of cumulative frequencies of the DTP and the PM for the TP7(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

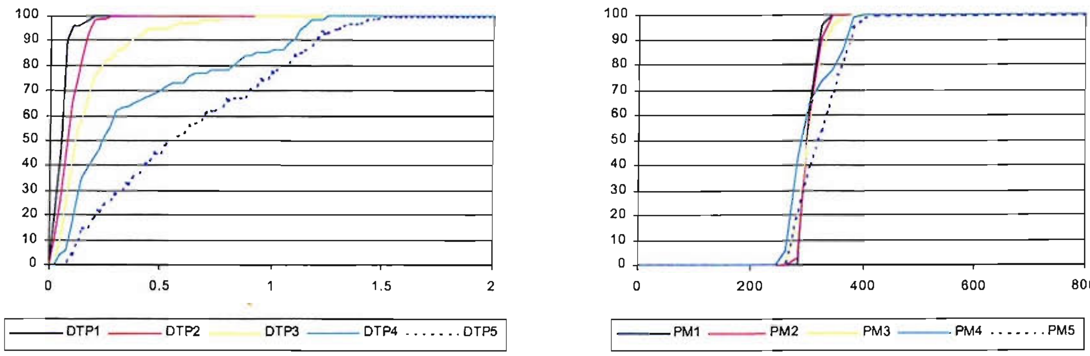


Figure 7.11: Curves of cumulative frequencies of the DTP and the PM for the TP7(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

From Figure 7.12 and Figure 7.13, the LCNM+PC method shown better performance than its original version from the standpoint of the accuracy and the PM. However, the LCNM method has better performance when the best local minimum is on the boundary rather than when it is located inside obtuse angle feasible region.

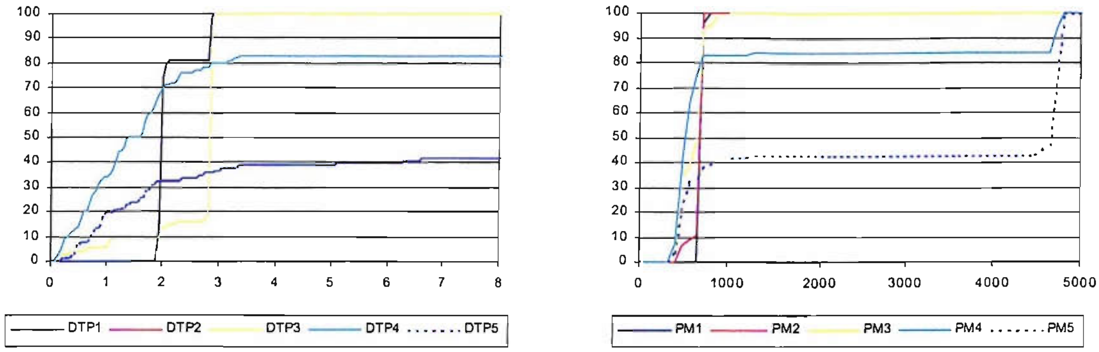


Figure 7.12: Curves of cumulative frequencies of the DTP and the PM for the TP8(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

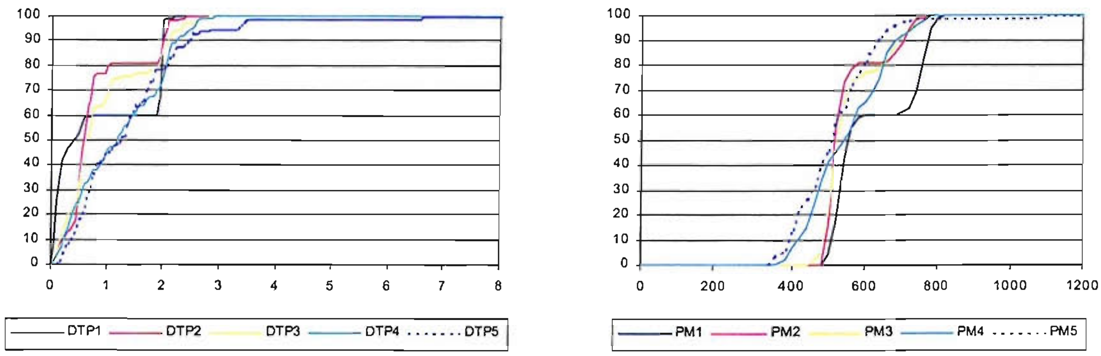


Figure 7.13: Curves of cumulative frequencies of the DTP and the PM for the TP8(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

As are shown in Figure 7.14 and Figure 7.15 the LCNM+PC method always converges close to the point $(1.3, 1.3, -1.8, 3.5)^T$ with a value of function approximately of 155 according to the report. In contrast, the LCNM method reaches the best local minimum in approximately 20% of the sample. From this viewpoint, one could say that the LCNM method is better than its modified version. However, the PM of the LCNM method is considerably greater than the PM of the LCNM+PC method.

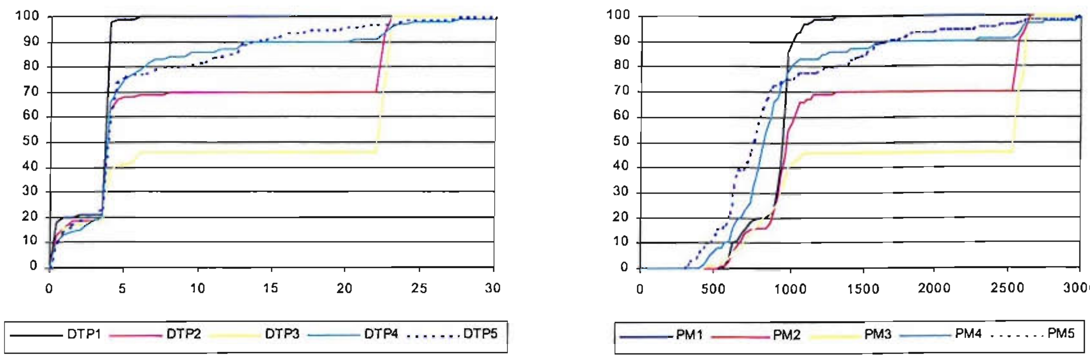


Figure 7.14: Curves of cumulative frequencies of the DTP and the PM for the TP17(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

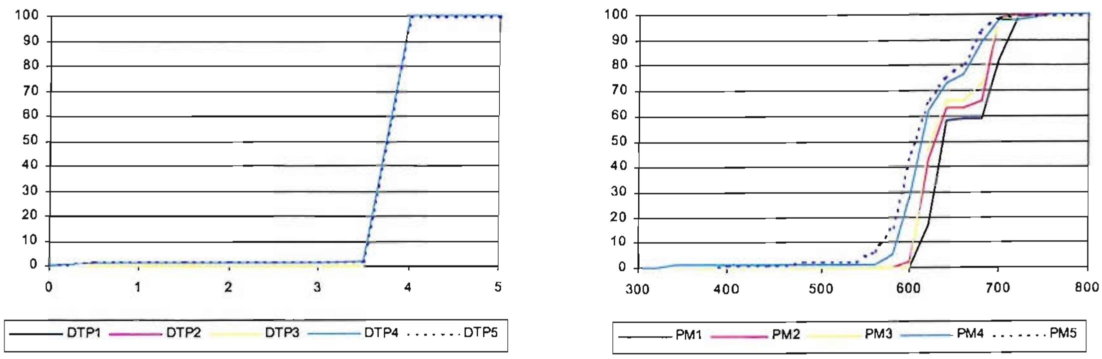


Figure 7.15: Curves of cumulative frequencies of the DTP and the PM for the TP17(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

From Figure 7.16 and Figure 7.17, we observe that the LCNM method has better performance than its modified version, considering the DTP and the PM. On the other hand, when we compare the performance of each method in the test problems TP17(4) and TP18(4), we note that both versions of the method converge with more frequency to points close to the true point, when the best local minimum is inside the feasible region.

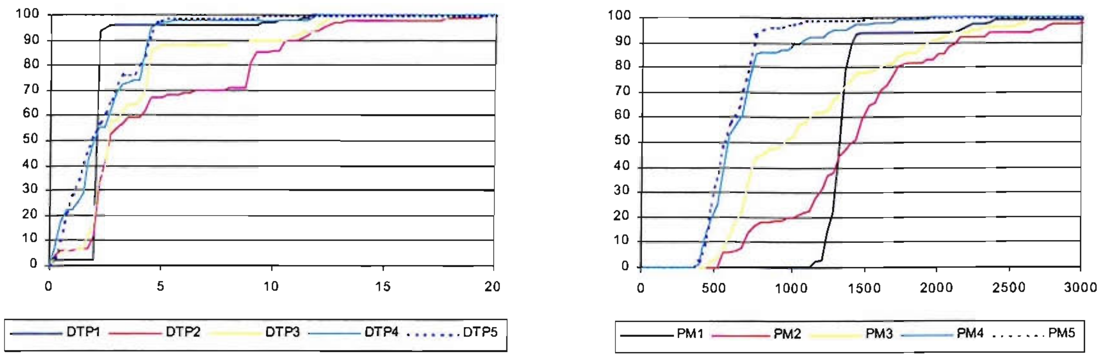


Figure 7.16: Curves of cumulative frequencies of the DTP and the PM for the TP18(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM.

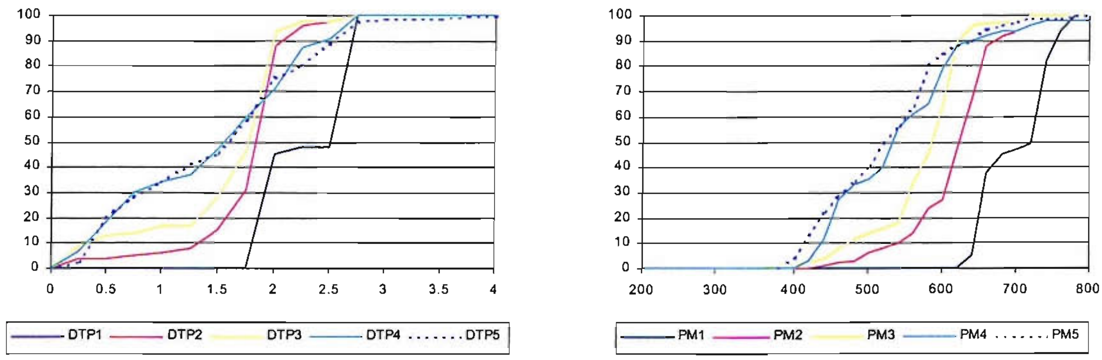


Figure 7.17: Curves of cumulative frequencies of the DTP and the PM for the TP18(4) with noise level $\sigma = 0.1$, 0.5, 1, 5 and 10, using the LCNM+PC.

7.5 Conclusions

From the comparative analysis developed in this chapter, we have recognized the goodness of both algorithms for estimating the optimum to constrained noisy minimization problem. Both algorithms identify a reasonable minimum within an adequate number of function evaluations.

The comparative study between our algorithms by 90th percentile shows that the LCNM+PC algorithm has better performance that its original version for both class of region and location of the constrained global minimum. Nevertheless, this fact cannot be generalized to the comparative analysis by median, where the LCNM algorithm seemed to have better performance, for 27 of 40 cases, than its premature collapse version when the feasible region has an obtuse solid angle. In contrast, when the feasible region is sharp, the LCNM+PC method identified more efficiently the minimum in the 75% of the cases than the LCNM method.

In our opinion, the LCNM+PC algorithm could be an useful approach rather than the LCNM algorithm, if it saved the information of the evaluated points for inferring and recognizing some features of the objective function during its current stage. We believe therefore that the LCNM+PC algorithm can make the decision whether to keep using the premature collapse criterion or not in the following stage of the algorithm. Thus the use of premature collapse criterion in any stage of the algorithm would depend on, for example, the possible number of valleys of the objective function that can be inferred and its probable location. Obviously, this recommendation would be an interesting research for improving the LCNM+PC algorithm.

Chapter 8

Conclusions and future research

Through this research we have developed an algorithm for optimizing a non-linear objective function subject to linear inequality constraints, when the analytical expression of the objective function is unavailable or its evaluation at each experimental design point is expensive. The LCNM algorithm and its premature collapse version have been shown to be able to identify at least a constrained local minimum within a reasonable number of function evaluations, even in situations when the initial point for building the simplex is far from the optimum. The algorithms have also been shown to have a satisfactory behaviour in linearly constrained optimization problems when the objective function is corrupted by noise.

An interesting result of the minimization problem of a two-dimension linear objective function subject to two linear constraints demonstrated that the LCNM algorithm can be very expensive, due to its exceptionally slow rate of convergence of order $\mathcal{O}(q^{-1/2})$. However, the premature collapse of the simplex has been shown to be an useful approach for improving the efficiency of the LCNM algorithm in this kind of problems. Nonetheless, the induction of collapse of the simplex can intensify the search of optima on the boundary of the feasible region rather than inside the region, so missing the identification of a possible constrained global optimum located inside the feasible region. Hence, the use of the premature collapse in the LCNM algorithm can be somewhat risky.

The development of the LCNM algorithm and its premature collapse modification allow us to understand the behaviour of the methods based on the NM algorithm, which induced us to propose future research for improving and extending the use of the LCNM algorithm.

An interesting open problem is the development of an optimization algorithm based on the LCNM algorithm for identifying the optimum to problems where the decision variables are discrete. This would result in a powerful optimization tool to find the optimum operation of system that can only be evaluated through discrete settings, such as the optimization of queue networks, where some decision variables can uniquely be fixed by discrete quantities for estimating their performance measure.

The study of the called dynamic LCNM algorithm allow us to appreciate the potentiality of applying the so-called dynamic NM operations, which concerns with the estimation of the best value of the parameter of the NM operations using some criteria, so the algorithm can obtain an optimum trial point at each iteration. The investigation of some criteria for estimating the best value of the NM parameter operations can constitute an important contribution.

Nowadays some modifications of the NM simplex algorithm have been developed for avoiding the convergence to a non-stationary point, for instance, Kelley (1999) and Price et al. (2002). However, in our opinion, these approaches are still not good enough, because they can be expensive. The search of an economic algorithm for minimizing non-linear objective function subject to linear constraints that guarantees its convergence to local minimum is to date an open problem, as is the theoretical study of convergence.

An extension of the linearly constrained linear programming problem of Section 4.4 to more dimensions can be an interesting research, where the study of conditions for avoiding a slow convergence rate of the LCNM algorithm can help to correct this potential weakness.

Obviously, the development of an extension of the LCNM algorithm to non-linear constraint minimization problems can also represent an interesting investigation in the field of the non-linear programming.

Appendix A

Optimality conditions

This appendix deals with the optimality condition for the linearly constrained minimization (LCM) problems. In particular, problems will be considered subject to linear inequality constraints for proving necessary conditions, without any convexity assumptions of the objective function. Furthermore, if the objective function satisfies some convexity assumptions, sufficient conditions of optimality can be established . The following definitions and theorems are based on the textbooks written by Bazaraa and Shetty (1979), and Nocedal and Wright (1999). However, this concise theory was adapted to our particular LCM problems.

A.1 Preliminary definitions

Definition A.1 (ε-neighbourhood around $\mathbf{x}$) *A set $N_\varepsilon(\mathbf{x})$ is said to be the ε-neighbourhood of $\mathbf{x}$ in a d-dimensional Euclidean space $\mathbb{R}^d$, if all its points $\mathbf{y} \in \mathbb{R}^d$ hold the inequality $\|\mathbf{y} - \mathbf{x}\| < \varepsilon$ for some $\varepsilon > 0$. This means, $N_\varepsilon(\mathbf{x}) = \{\mathbf{y} \mid \|\mathbf{y} - \mathbf{x}\| < \varepsilon\}$.*

Definition A.2 (Point in closure of B) *Let B be an arbitrary and non-empty set in $\mathbb{R}^d$ and let cl B denote a set of points in the closure of B. A point $\mathbf{x}$ is said to be in cl B if $B \cap N_\varepsilon(\mathbf{x}) \neq \varnothing$ for every $\varepsilon > 0$.*

Definition A.3 (Interior point of B) *Let B be an arbitrary and non-empty set in $\mathbb{R}^d$ and let int B denote a set of points in the interior of B. A point $\mathbf{x}$ is said to be in int B if $N_\varepsilon(\mathbf{x}) \subset B$ for some $\varepsilon > 0$.*

Definition A.4 (Point in the boundary) *Let B be an arbitrary and non-empty set* *in* $\mathbb{R}^d$ *and let bnd B denote a set of points in the boundary of B. A point* $\mathbf{x}$ *is said to be in bnd B if* $N_\varepsilon(\mathbf{x}) \cap B \neq \varnothing$ *and* $N_\varepsilon(\mathbf{x}) \not\subseteq B$ *for every* $\varepsilon > 0$.

Definition A.5 (Redundant constraint) *Let* $\mathcal{F}$ *be a feasible region constituted by a set of* C_i *constraints for all* $i \in \Omega$. *A constraint* C_k *for a* $k \in \Omega$ *is said to be redundant if its removal does not modify the feasible region.*

Definition A.6 (Cone of feasible direction) *Let B be an arbitrary and non-empty set in* $\mathbb{R}^d$ *and let* $\mathbf{x}^*$ *be a point in cl B. A set* $D(\mathbf{x}^*)$ *is said to be the cone of feasible directions of B at* $\mathbf{x}^*$ *if*

$$\mathcal{D}_0(\mathbf{x}^*) = \{\mathbf{d} \mid \mathbf{d} \neq \mathbf{0} \wedge \mathbf{x}^* + \mu\mathbf{d} \in B\} \quad \forall 0 < \mu < \delta \text{ for some } \delta > 0,$$

where each non-zero vector $\mathbf{d} \in \mathcal{D}_0(\mathbf{x}^*)$ *is called a feasible direction.*

Definition A.7 (Convex function) *Let* $f(\mathbf{x}) : D \to \mathbb{R}$, *where D is a non-empty convex set in* $\mathbb{R}^d$. *We say that the function* $f(\mathbf{x})$ *is convex on D if* $f(\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2) \leq \mu\, f(\mathbf{x}_1) + (1-\mu)\, f(\mathbf{x}_2)$ *for each* $\mathbf{x}_1, \mathbf{x}_2 \in D$ *and for each* $0 \leq \mu \leq 1$.

Definition A.8 (Quasiconvex function) *Let* $f(\mathbf{x}) : D \to \mathbb{R}$, *where D is a non-empty convex set in* $\mathbb{R}^d$. *The function* $f(\mathbf{x})$ *is said to be quasiconvex on D if* $f(\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2) \leq \max\{f(\mathbf{x}_1),\, f(\mathbf{x}_2)\}$ *for each* $\mathbf{x}_1, \mathbf{x}_2 \in D$ *and for each* $0 < \mu < 1$.

Definition A.9 (Quasiconcave function) *Let* $f(\mathbf{x}) : D \to \mathbb{R}$, *where D is a non-empty convex set in* $\mathbb{R}^d$. *The function* $f(\mathbf{x})$ *is said to be quasiconcave on D if* $f(\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2) \geq \min\{f(\mathbf{x}_1),\, f(\mathbf{x}_2)\}$ *for each* $\mathbf{x}_1, \mathbf{x}_2 \in D$ *and for each* $0 < \mu < 1$.

Definition A.10 (Pseudoconvex function) *Let D be a non-empty open set in* $\mathbb{R}^d$, *and let* $f(\mathbf{x}) : D \to \mathbb{R}$ *be a differentiable function on D. The function* $f(\mathbf{x})$ *is said to be pseudoconvex if for each* $\mathbf{x}_1 \in D$, $\mathbf{x}_2 \in D$ *and the scalar product of* $\nabla f(\mathbf{x}_1)^T(\mathbf{x}_2 - \mathbf{x}_1) \geq 0$, *then* $f(\mathbf{x}_2) \geq f(\mathbf{x}_1)$, *or if* $f(\mathbf{x}_2) < f(\mathbf{x}_1)$ *then* $\nabla f(\mathbf{x}_1)^T(\mathbf{x}_2 - \mathbf{x}_1) < 0$.

Definition A.11 (Differentiable function) *Let B be an arbitrary and non-empty set in* $\mathbb{R}^d$. *We say that* $f(\mathbf{x}) : B \to \mathbb{R}$ *is differentiable at* $\mathbf{x}_0$ *in int B if there exists a gradient vector, denoted by* $\nabla f(\mathbf{x}_0)$, *and a function* $\alpha(\mathbf{x}_0, \mathbf{x} - \mathbf{x}_0) : \mathbb{R}^d \to \mathbb{R}$, *such that*

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0) + \|\mathbf{x} - \mathbf{x}_0\|\, \alpha(\mathbf{x}_0, \mathbf{x} - \mathbf{x}_0) \quad \forall \mathbf{x}_0 \in B,$$

where $\alpha(\mathbf{x}_0, \mathbf{x} - \mathbf{x}_0) \to 0$ *as* $\mathbf{x} \to \mathbf{x}_0$.

Definition A.12 (Global and local minimum point) *Let* $\mathcal{X}$ *be a non-empty open set in* $\mathbb{R}^d$ *and let* $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$. *Consider Problem* $\mathcal{P}$ *of minimizing* $f(\mathbf{x})$ *subject to* $\mathbf{x} \in (\mathcal{X} \cap D)$. *If a point* $\mathbf{x}^* \in D$ *is such that* $f(\mathbf{x}^*) \leq f(\mathbf{x})$ *for all* $\mathbf{x} \in D$, *then* $\mathbf{x}^*$ *is called global minimum point to Problem* $\mathcal{P}$. *If a point* $\mathbf{x}^* \in D$ *and if there exists an* ε*-neighbourhood of* $\mathbf{x}^*$ *such that* $f(\mathbf{x}^*) \leq f(\mathbf{x})$ *for all* $\mathbf{x} \in (D \cap N_\varepsilon(\mathbf{x}^*))$, *then* $\mathbf{x}^*$ *is considered a local minimum point to Problem* $\mathcal{P}$.

It is worthwhile mentioning that the minimum points can also be called a constrained global minimum and a constrained local minimum, when they are the global minimum and a local minimum to a constrained minimization problem, respectively. Similarly, if Problem $\mathcal{P}$ were an unconstrained minimization problem, the minimum could be called an unconstrained global minimum and unconstrained local minimum.

A.2 Preliminary theorems

Lemma A.1 *Let* $l_i(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ *be a linear function given by* $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$, *and* $\mathcal{F} = \{l_i(\mathbf{x}) = a_i^T\mathbf{x} - b_i \geq 0 \mid i \in \mathcal{I}\}$ *be a non-empty set given by linear inequality constraints, where* $\mathcal{I}$ *is the set of the subscripts of the linear inequality constraints. Then each linear function* $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ *is a quasiconcave function.*

Proof. Let $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^d$ and $\mu \in \mathbb{R}$ such that $0 < \mu < 1$. Thus, for each $i \in \mathcal{I}$,

$$l_i[\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = \mathbf{a}_i^T[\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2] - b_i \tag{A.1}$$

After some calculations, Equation (A.1) is rewritten as

$$l_i[\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = \mu\ l_i(\mathbf{x}_1) + (1-\mu)\ l_i(\mathbf{x}_2) \tag{A.2}$$

Now, suppose that $l_i(\mathbf{x}_1) \geq l_i(\mathbf{x}_2)$, this implies, $l_i(\mathbf{x}_1) = l_i(\mathbf{x}_2) + \Delta$, where $\Delta \geq 0$, so Equation (A.2) can be written as

$$l_i[\mu\mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = \mu\ l_i(\mathbf{x}_2) + \mu\Delta + (1-\mu)\ l_i(\mathbf{x}_2).$$

Therefore,

$$l_i[\mu \mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = l_i(\mathbf{x}_2) + \mu\Delta \geq l_i(\mathbf{x}_2) = \min\{l_i(\mathbf{x}_1), l_i(\mathbf{x}_2)\} \tag{A.3}$$

Furthermore, suppose that $l_i(\mathbf{x}_2) \geq l_i(\mathbf{x}_1)$, this implies, $l_i(\mathbf{x}_2) = l_i(\mathbf{x}_1) + \Delta$, where $\Delta \geq 0$, then Equation (A.2) is rewritten as

$$l_i[\mu \mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = \mu \; l_i(\mathbf{x}_1) + (1-\mu) \; l_i(\mathbf{x}_1) + (1-\mu)\Delta.$$

As a result of the above equation, we obtain

$$l_i[\mu \mathbf{x}_1 + (1-\mu)\mathbf{x}_2] = l_i(\mathbf{x}_1) + (1-\mu)\Delta \geq l_i(\mathbf{x}_1) = \min\{l_i(\mathbf{x}_1), l_i(\mathbf{x}_2)\} \tag{A.4}$$

Both Equations (A.3) and (A.4) satisfy the definition of quasiconcave function. ■

Theorem A.1 (**Theorem of Gordan**) *Let A be an $m \times n$ matrix. If System S_1 is given by $A\mathbf{x} < \mathbf{0}$ for some $\mathbf{x}$ in $\mathbb{R}^n$ and System S_2 is defined by $A^T\mathbf{u} = \mathbf{0}$ and $\mathbf{u} \geq \mathbf{0}$ for some non-zero $\mathbf{u}$ in $\mathbb{R}^m$, then there exists one unique solution.*

Proof. See (Mangasarian 1969). ■

Note that if the feasible region is defined by linear constraints only, the Mangasarian Fromowitz constraint qualification is automatically true.

A.3 The problem

Consider the following linearly constrained minimization problem $\mathcal{P}$

Problem A.1 ($\mathcal{P}$) *Let $\mathcal{P}$ be a linearly constrained minimization problem given by*

$$\min_{\mathbf{x} \in \mathbb{R}^d} f(\mathbf{x})$$

subject to

$$l_i(\mathbf{x}) \geq 0 \quad \forall i \in \mathcal{I}$$

where $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ is a linear function $\mathbb{R}^d \to \mathbb{R}$ for every $i \in \mathcal{I}$ and $\mathcal{I}$ is the set of the subscripts of the linear inequality constraints.

Let $\mathcal{A}_p(\mathbf{x}_0)$ define the set of subscripts $i \in \mathcal{I}$ such that $l_i(\mathbf{x}_0) = 0$, that is, the active inequality constraints due to $\mathbf{x}_0$ and it can be written as $\mathcal{A}_p(\mathbf{x}_0) = \{i \in \mathcal{I} | l_i(\mathbf{x}_0) = 0\}$.

A.4 Conditions of optimality

In this section, necessary optimality conditions will be given for a generalized constrained minimization problem for linearly constrained minimization problems. Finally, sufficient conditions of optimality will be developed under some convexity assumptions.

Theorem A.2 (Descent direction of $f(\mathbf{x})$) *Let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function at $\mathbf{x}_0$. If there exists a vector $\mathbf{d} \in \mathbb{R}^d$ such that $\nabla f(\mathbf{x}_0)^T\mathbf{d} < 0$, then there exists a $\delta > 0$ such that $f(\mathbf{x}_0 + \mu\mathbf{d}) < f(\mathbf{x}_0)$ $\forall 0 < \mu < \delta$. This means that $\mathbf{d}$ is a descent direction of $f(\mathbf{x})$ at the point $\mathbf{x}_0$.*

Proof. Since $f(\mathbf{x})$ is differentiable at the point $\mathbf{x}^*$, we say

$$f(\mathbf{x}) = f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0)^T(\mathbf{x} - \mathbf{x}_0) + \|\mathbf{x} - \mathbf{x}_0\| \, \alpha(\mathbf{x}_0, \mathbf{x} - \mathbf{x}_0).$$

If $\mathbf{x} = \mathbf{x}_0 + \mu\mathbf{d}$ then $\mathbf{x} - \mathbf{x}_0 = \mu\mathbf{d}$, and

$$f(\mathbf{x}_0 + \mu\mathbf{d}) = f(\mathbf{x}_0) + \mu\nabla f(\mathbf{x}_0)^T\mathbf{d} + \mu \|\mathbf{d}\| \, \alpha(\mathbf{x}_0, \mu\mathbf{d}),$$

where $\alpha(\mathbf{x}_0, \mu\mathbf{d}) \to 0$ as $\mu \to 0$.

By rearranging the terms, we obtain

$$\frac{f(\mathbf{x}_0 + \mu\mathbf{d}) - f(\mathbf{x}_0)}{\mu} = \nabla f(\mathbf{x}_0)^T\mathbf{d} + \|\mathbf{d}\| \, \alpha(\mathbf{x}_0, \mu\mathbf{d}).$$

Due to the fact that $\nabla f(\mathbf{x}_0)^T\mathbf{d} < 0$ and $\alpha(\mathbf{x}_0, \mu\mathbf{d}) \to 0$ as $\mu \to 0$, there exists a $\delta > 0$ such that

$$\nabla f(\mathbf{x}_0)^T\mathbf{d} + \|\mathbf{d}\| \, \alpha(\mathbf{x}_0, \mu\mathbf{d}) < 0.$$

Therefore, $f(\mathbf{x}_0 + \mu\mathbf{d}) < f(\mathbf{x}_0)$ which means that $\mathbf{d}$ is a descent direction of $f(\mathbf{x})$ at the point $\mathbf{x}_0$. ∎

Theorem A.3 (Local minimum of $f(\mathbf{x})$) *Let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a differentiable function at a point $\mathbf{x}^* \in B$. If $\mathbf{x}^*$ locally solves a problem of minimization of $f(\mathbf{x})$ subject* *to $\mathbf{x} \in B$, then $F_0(\mathbf{x}^*) \cap \mathcal{D}_0(\mathbf{x}^*) = \varnothing$, where $F_0(\mathbf{x}^*) = \{\mathbf{d} \mid \nabla f(\mathbf{x}^*)^T \mathbf{d} < 0\}$ is the cone of descent direction of $f(\mathbf{x})$ and $\mathcal{D}_0(\mathbf{x}^*)$ is the cone of feasible directions of B at $\mathbf{x}^*$.*

Proof. According to the hypothesis of the theorem, $\mathbf{x}^*$ is a local minimum, this means that there exists a non-descent and non-zero vector direction $\mathbf{d}$ such that minimizes $\mathbf{x}^*$ in an ε-neighbourhood around $\mathbf{x}^*$.

Thus, if $\mathbf{x}^*$ is a local minimum, then $f(\mathbf{x}^* + \mu\mathbf{d}) > f(\mathbf{x}^*)$ for $0 < \mu < \delta_1$, where $\mathbf{d} \in \mathcal{D}_0(\mathbf{x}^*)$.

Furthermore, according to Theorem A.2, $F_0(\mathbf{x}^*) = \{\mathbf{d} \mid \nabla f(\mathbf{x}^*)^T \mathbf{d} < 0\}$ is the set of directions that minimizes $f(\mathbf{x})$.

On the other hand, the set of minimization directions $F_0(\mathbf{x}^*)$ uniquely contains non-zero descent direction vectors. Therefore, there does not exist a non-zero vector that belongs to $\mathcal{D}_0(\mathbf{x}^*)$ and $F_0(\mathbf{x}^*)$ simultaneously, which allows one to assure that $F_0(\mathbf{x}^*) \cap \mathcal{D}_0(\mathbf{x}^*) = \varnothing$. ■

Theorem A.4 (Local minimum for the LCM problems) *Let $\mathcal{X}$ be a non-empty open set in $\mathbb{R}^d$ and let $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ and $l_i(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ be a set of linear functions given by $l_i(\mathbf{x}) = \mathbf{a}_i^T \mathbf{x} - b_i$ for all $i \in \mathcal{I}$. Consider Problem $\mathcal{P}$ of minimizing $f(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{X}$ and $l_i(\mathbf{x}) \geq 0$ for every $i \in \mathcal{I}$. If $\mathbf{x}^*$ locally solve Problem $\mathcal{P}$ and there exists a non-empty set $\mathcal{A}_p(\mathbf{x}^*) = \{i \in \mathcal{I} | l_i(\mathbf{x}^*) = 0\}$, then $F_0(\mathbf{x}^*) \cap G_0(\mathbf{x}^*) = \varnothing$, where each non-zero vector direction $\mathbf{d}$ holds*

$$F_0(\mathbf{x}^*) = \{\mathbf{d} \mid \nabla f(\mathbf{x}^*)^T \mathbf{d} < 0\}$$

$$G_0(\mathbf{x}^*) = \{\mathbf{d} \mid \mathbf{a}_i^T \mathbf{d} > 0 \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*)\}$$

Proof. Suppose that there exists a non-zero vector $\mathbf{d}_0 \in F_0(\mathbf{x}^*)$, whereby $\mathbf{d}_0$ is a descent direction of $f(\mathbf{x})$ at the point $\mathbf{x}^*$, because the angle between $\mathbf{d}_0$ and $\nabla f(\mathbf{x})$ is obtuse. This condition cannot simultaneously satisfy the fact that $\mathbf{a}_i^T \mathbf{d}_0 > 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$, where $\mathbf{a}_i$ is the normal vector of the boundary of the constraint $l_i(\mathbf{x}^*) \geq 0$. It is evident that if $\nabla^T f(\mathbf{x}^*)\mathbf{d}_0 < 0$, then $\mathbf{d}_0$ cannot belong to $G_0(\mathbf{x}^*)$, because there exists at least a kth ($k \in \mathcal{A}_p(\mathbf{x}^*)$) active constraint whose $\mathbf{a}_k^T \mathbf{d}_0 < 0$, which is inconsistent with the definition of $G_0(\mathbf{x}^*)$. ■

Now we show the necessary conditions of optimality of Fritz John, and the necessary conditions of optimality of Kuhn-Tucker (KT). This latter is often known as the First-Order necessary condition of Karush-Kuhn-Tucker (KKT) (Nocedal and Wright 1999).

Theorem A.5 (The Fritz John conditions for the LCM problems) *Let $\mathcal{X}$ be a non-empty open set in $\mathbb{R}^d$. Let $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a non-linear function and let $l_i(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be a set of linear functions given by $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ for all $i \in \mathcal{I}$. Consider Problem $\mathcal{P}$ of minimizing $f(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{X}$ and $l_i(\mathbf{x}) \geq 0$ for every $i \in \mathcal{I}$. Let $\mathbf{x}^*$ be a feasible point such that $\mathbf{x}^*$ lies on all active linear ith constraints , this is, i belongs to a non-empty set $\mathcal{A}_p(\mathbf{x}^*) = \{i \in \mathcal{I} | l_i(\mathbf{x}^*) = 0\}$. Furthermore, suppose that $f(\mathbf{x})$ is differentiable at $\mathbf{x}^*$. If $\mathbf{x}^*$ is a local minimum, then there exist a set of scalars u_0 and u_i for all $i \in \mathcal{A}_p(\mathbf{x}^*)$, such that*

$$\begin{aligned} & u_0 \nabla f(\mathbf{x}^*) - \textstyle\sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} u_i \mathbf{a}_i = \mathbf{0} \\ & u_0, u_i \geq 0 \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*) \\ & (u_0, \mathbf{u}_\mathcal{A}) \neq (0, \mathbf{0}) \end{aligned} \tag{A.5}$$

Proof. Due to the fact that $\mathbf{x}^*$ locally solves Problem $\mathcal{P}$, there does not exist a non-zero vector $\mathbf{d}$ such that simultaneously hold $\nabla^T f(\mathbf{x}^*)\mathbf{d} < 0$ and $\mathbf{a}_\mathrm{i}^\mathbf{T}\mathbf{d} > 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$. This latter inequality is equivalent to $-\mathbf{a}_\mathrm{i}^\mathbf{T}\mathbf{d} < 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$. According to Theorem A.4 the condition of optimality can be written as

$$\begin{bmatrix} \nabla f(\mathbf{x}^*)^T \\ -A_\mathcal{A} \end{bmatrix} \mathbf{d} < \mathbf{0} \tag{A.6}$$

where $A_\mathcal{A}$ is the matrix whose rows are $\mathbf{a}_\mathrm{i}^\mathbf{T}$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$.

Since Inequality A.6 is inconsistent due to Theorem A.1, there exists a non-zero vector $(u_0, \mathbf{u}_\mathcal{A}) \geq \mathbf{0}$ such that

$$u_0 \nabla f(\mathbf{x}^*) - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} u_i \mathbf{a}_i = \mathbf{0}. \tag{A.7}$$

Due to the conditions of Theorem A.1, we can establish that

$$\begin{aligned} (u_0, \mathbf{u}_\mathcal{A}) \geq \mathbf{0} \\ (u_0, \mathbf{u}_\mathcal{A}) \neq \mathbf{0} \end{aligned}$$

where the scalars u_0 and u_i for all $i \in \mathcal{I}$ are widely known as Lagragian multipliers. ■

Note that the original Theorem of Fritz assumes the differentiability of $l_i(\mathbf{x})$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$ and the continuity of $l_i(\mathbf{x})$ for all $i \notin \mathcal{A}_p(\mathbf{x}^*)$ (Bazaraa and Shetty 1979), which were omitted, because all linear functions are obviously differentiable and continuous.

The Theorem of Fritz is a necessary but non-sufficient conditions of optimality, this means, all local minimum points must satisfy Equation (A.5). However, there could exist non-optimum points that satisfy Equation (A.5). Nevertheless, if some $\mathbf{x}_c$ point does not satisfy Fritz John conditions, we can be assured that $\mathbf{x}_c$ is non-local minimum.

Lemma A.2 (Normal of active linear inequality constraints) *Let $l_i(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be linear functions given by $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$, and $\mathcal{F} = \{l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i \geq 0 \mid i \in \mathcal{I}\}$ be a non-empty set given by k non-redundant linear inequality constraints. If a point $\mathbf{x}_0$ actives a subset of linear inequality constraints of $\mathcal{F}$, this is, there exists a non-empty set $\mathcal{A}_p(\mathbf{x}_0) = \{i \in \mathcal{I} | l_i(\mathbf{x}_0) = a_i^T\mathbf{x}_0 - b_i = 0\}$, then every normal $\mathbf{a}_i$ for $i \in \mathcal{A}_p(\mathbf{x}_0)$ are linearly independent.*

Proof. Because there exists at least one feasible point $\mathbf{x}$, such that it activates m ($\leq k$) non-redundant linear constraints of $\mathcal{F}$, then the linear system given by the active constraints can be written as,

$$A\mathbf{x} - \mathbf{b} = \mathbf{0}, \tag{A.8}$$

where A is an $(m \times d)$-dimensional non-zero matrix, $\mathbf{x}$ is a d-dimensional vector and, $\mathbf{b}$ and $\mathbf{0}$ are m-dimensional vectors.

Since there exist non-redundant linear constraints, matrix A and vector $\mathbf{x}$ can be arranged by simple simultaneous permutating columns of matrix A and permutating rows of vector $\mathbf{x}$ only, to obtain an equivalent linear system given by partitioned submatrices: A_R of dimension $m \times m$ and rank m, and A_N of dimension $m \times (d - m)$. That is, Equation (A.8) can be rewritten as

$$[A_R \ A_N] \begin{bmatrix} \mathbf{x}_R \\ \mathbf{x}_N \end{bmatrix} = \mathbf{b} \tag{A.9}$$

where $\mathbf{x}_R$ is an m-dimensional vector and $\mathbf{x}_N$ is a $(d - m)$-dimensional vector.

Since the rank of A_R is equal to m, then the row rank of $[A_R \ A_N]$ is equal to m. Hereby, the linear system given by Equation (A.8) is conformed by m linear independent equations. Hence all $\mathbf{a}_i$ normal for $i \in \mathcal{A}_p(\mathbf{x}_0)$ are linearly independent. ∎

It is worthwhile pointing out that if the feasible region of Problem $\mathcal{P}$ is defined by non-redundant linear inequality constraints, then the normals $\mathbf{a}_i$ for $i \in \mathcal{A}_p(\mathbf{x}^*)$ are linearly independent (Lemma A.2), which satisfies one of the assumptions of KT necessary optimality conditions. Furthermore, since all linear functions are continuous and differentiable, Theorem of KT necessary conditions for LCM problems can be enunciated as follows.

Theorem A.6 (Kuhn-Tucker necessary conditions for the LCM problems) *Let $\mathcal{X}$ be a non-empty open set in $\mathbb{R}^d$ and let $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ and $l_i(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ be a set of linear functions given by $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ for all $i \in \mathcal{I}$. Consider Problem $\mathcal{P}$ of minimizing $f(\mathbf{x})$ subject to $\mathbf{x} \in \mathcal{X}$ and $l_i(\mathbf{x}) \geq 0$ for every $i \in \mathcal{I}$. Let $\mathbf{x}^*$ be a feasible point such that $\mathbf{x}^*$ lies on all active linear ith constraints , this is, i belongs to a non-empty set $\mathcal{A}_p(\mathbf{x}^*) = \{i \in \mathcal{I} | l_i(\mathbf{x}^*) = 0\}$. Furthermore, suppose that $f(\mathbf{x})$ is differentiable at $\mathbf{x}^*$. If $\mathbf{x}^*$ is a local minimum, then there exist scalars $\tilde{u}_i$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$, such that*

$$\begin{aligned} &\nabla f(\mathbf{x}^*) - \textstyle\sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \mathbf{a}_i = \mathbf{0} \\ &\tilde{u}_i \geq 0 \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*) \end{aligned} \tag{A.10}$$

Proof. Because of Theorem A.5, there exist scalars $u_0 \geq 0$ and $u_i \geq 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$ such that

$$u_0 \nabla f(\mathbf{x}^*) - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} u_i \mathbf{a}_i = \mathbf{0} \tag{A.11}$$

$$(u_0, \mathbf{u}_{\mathcal{A}}) \neq (0, \mathbf{0}) \tag{A.12}$$

Under the condition of linear independence of all normals $\mathbf{a_i}$ for $i \in \mathcal{A}_p(\mathbf{x}^*)$, u_0 must be positive, because $\sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} u_i \mathbf{a}_i \neq \mathbf{0}$, which evidently satisfies Equation (A.12). Therefore, letting $\tilde{u}_i = u_i / u_0$ Equation A.11 can be rewritten as

$$\nabla f(\mathbf{x}^*) - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \mathbf{a}_i = \mathbf{0},$$

where $\tilde{u}_i \geq 0$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$. ∎

Now, under some convexity assumptions, the KT conditions become sufficient conditions of global optimality for Problem $\mathcal{P}$. However, Nocedal and Wright (1999) present sufficient conditions of Karush-Kuhn-Tucker (KKT), which is founded on the existence of the second derivatives of $f(\mathbf{x})$, among others conditions, which are stronger than the KT conditions

Theorem A.7 (Kuhn-Tucker sufficient conditions for the LCM problems) *Let $\mathcal{X}$ be a non-empty open set in $\mathbb{R}^d$ and let $f(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ and $l_i(\mathbf{x}) : \mathbb{R}^d \rightarrow \mathbb{R}$ be a set of linear functions given by $l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i$ for all $i \in \mathcal{I}$. Consider Problem $\mathcal{P}$ of* *minimizing* $f(\mathbf{x})$ *subject to* $\mathbf{x} \in \mathcal{X}$ *and* $\mathcal{F} = \{l_i(\mathbf{x}) = \mathbf{a}_i^T\mathbf{x} - b_i \geq 0 \mid i \in \mathcal{I}\}$. *Let* $\mathbf{x}^*$ *be a feasible point such that* $\mathbf{x}^*$ *lies on all active linear ith constraints , this is, i belongs to a non-empty set* $\mathcal{A}_p(\mathbf{x}^*) = \{i \in \mathcal{I} | l_i(\mathbf{x}^*) = 0\}$. *If* $f(\mathbf{x})$ *is pseudoconvex and differentiable at* $\mathbf{x}^*$ *and Kuhn-Tucker necessary conditions are satisfied at* $\mathbf{x}^*$, *then* $\mathbf{x}^*$ *is a global minimum to Problem* $\mathcal{P}$.

Proof. Let $\mathbf{x}$ be a feasible point to Problem $\mathcal{P}$ such that for $i \in \mathcal{A}_p(\mathbf{x}^*)$, $l_i(\mathbf{x}) \geq 0$. Since each $l_i(\mathbf{x})$ for $i \in \mathcal{I}$ is quasiconcave (Lemma A.1), we have that for all $0 < \mu < 1$,

$$l_i[\mathbf{x}^* + \mu(\mathbf{x} - \mathbf{x}^*)] = l_i[\mu\mathbf{x} + (1-\mu)\mathbf{x}^*] \geq \min\{l_i(\mathbf{x}), l_i(\mathbf{x}^*)\} = l_i(\mathbf{x}^*) = 0 \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*). \tag{A.13}$$

Since $l_i(\mathbf{x})$ is a linear function and clearly differentiable,

$$\frac{l_i[\mathbf{x}^* + \mu(\mathbf{x} - \mathbf{x}^*)] - l_i(\mathbf{x}^*)}{\mu} = \nabla^T l_i(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*) \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*). \tag{A.14}$$

Because $\nabla l_i(\mathbf{x}^*) = \mathbf{a}_i$ for all $i \in \mathcal{A}_p(\mathbf{x}^*)$, $\mu > 0$ and Equation (A.13), Equation (A.14) is expressed as,

$$\mathbf{a}_i^T(\mathbf{x} - \mathbf{x}^*) \geq 0 \quad \forall i \in \mathcal{A}_p(\mathbf{x}^*) \tag{A.15}$$

On the other hand, if $f(\mathbf{x})$ is pseudoconvex and differentiable at $\mathbf{x}^*$, then

$$\nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \geq 0 \text{ and } f(\mathbf{x}) \geq f(\mathbf{x}^*) \tag{A.16}$$

Since $\mathbf{x}^*$ satisfies the necessary conditions of KT, Equation (A.10) can be written by its transpose

$$\nabla f(\mathbf{x}^*)^T - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \mathbf{a}_i^T = \mathbf{0}^T \tag{A.17}$$

By postmultiplying both sides of Equation (A.17), we obtain

$$\nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) - \sum_{i \in \mathcal{A}_p(\mathbf{x}^*)} \tilde{u}_i \mathbf{a}_i^T(\mathbf{x} - \mathbf{x}^*) = 0,$$

which is satisfied by Equations (A.15) and (A.16). ■

A.5 Examples

In this section we illustrate the previous theory with some examples that have been employed in this work, and verify the optimality of some points that were assumed as minima in the reported test problems.

A.5.1 Test Problem 1

$$\min_{\mathbf{x}\in\mathbb{R}^d} \sum_{i=1}^{d} x_i^2$$

subject to

$$C_1:\ \sum_{i=1}^{d} x_i \geq 3$$
$$C_2:\ 2x_1 + \sum_{i=2}^{d} x_i \geq 5$$

The points for being verified its optimality conditions are

$$\mathbf{x}_{local}^T = \begin{cases} [2,1] & d=2, \\ [1.42857,\ 0.714286,\ 0.714286,\ 0.714286] & d=4, \\ [1.11111,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555,\ 0.55555] & d=6. \end{cases}$$

Case $d = 2$

At $\mathbf{x}^* = (2,1)^T$ we have $\mathcal{A}_p(\mathbf{x}^*) = \{1,2\}$

$$\nabla f(\mathbf{x}^*)^T = (4,2) \quad \mathbf{a}_1^T = (1,1) \quad \mathbf{a}_2^T = (2,1)$$

Using Equation (A.10), we obtain

$$\begin{bmatrix} 4 \\ 2 \end{bmatrix} - \tilde{u}_1 \begin{bmatrix} 1 \\ 1 \end{bmatrix} - \tilde{u}_2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \mathbf{0}$$

which is satisfied for $\tilde{u}_1 = 0$ and $\tilde{u}_2 = 2$. Since $f(\mathbf{x})$ is pseudoconvex, $(2,1)^T$ is the constrained global minimum.

Case $d = 4$

At $\mathbf{x}^* = (1.42857,\ 0.714286,\ 0.714286,\ 0.714286)^T$ we have $\mathcal{A}_p(\mathbf{x}^*) = \{2\}$

$$\nabla f(\mathbf{x}^*)^T = (2.8571,\ 1.4286,\ 1.4286,\ 1.4286\ ),\quad \mathbf{a}_2^T = (2,1,1,1)$$

Thus,

$$\begin{bmatrix} 2.8571 \\ 1.4286 \\ 1.4286 \\ 1.4286 \end{bmatrix} - \tilde{u}_2 \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \mathbf{0}$$

The above equation is clearly satisfied for $\tilde{u}_2 = 1.4286$. Since $f(\mathbf{x})$ is pseudoconvex, $\mathbf{x}^*$ is the constrained global minimum.

Case $d = 6$

At $\mathbf{x}^* = (1.1111,\ 0.5555,\ 0.5555,\ 0.5555,\ 0.5555,\ 0.5555)^T$ we have $\mathcal{A}_p(\mathbf{x}^*) = \{2\}$

$$\nabla f(\mathbf{x}^*)^T = (2.2222,\ 1.1111,\ 1.1111,\ 1.1111,\ 1.1111,\ 1.1111\ )\quad \mathbf{a}_2^T = (2,1,1,1,1,1)$$

Thus,

$$\begin{bmatrix} 2.2222 \\ 1.1111 \\ 1.1111 \\ 1.1111 \end{bmatrix} - \tilde{u}_2 \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \mathbf{0}$$

when $\tilde{u}_2 = 1.1111$. Since $f(\mathbf{x})$ is pseudoconvex, $\mathbf{x}^*$ is the constrained global minimum.

A.5.2 Test Problem 5

Case $d = 2$

$$\min_{\mathbf{x}\in\mathbb{R}^d} [100(x_2 - x_1^2)^2 + (1 - x_1)^2]$$

subject to

$$C_1:\ \ x_1 + x_2 \geq 3$$
$$C_2:\ \ 2x_1 + x_2 \geq 5$$

At $\mathbf{x}^* = (-3.447634,\ 11.895268)^T$ we have $\mathcal{A}_p(\mathbf{x}^*) = \{2\}$

$$\nabla f(\mathbf{x}^*)^T = (3.6373,\ 1.8176) \quad \mathbf{a}_2^T = (2,1)$$

Therefore,

$$\begin{bmatrix} 3.6373 \\ 1.8176 \end{bmatrix} - \tilde{u}_2 \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \mathbf{0}$$

The above equation holds for $\tilde{u}_2 = 1.8176$. This implies that the solution satisfies the necessary conditions of KT. Since the objective function clearly is pseudoconvex in $N_\varepsilon(\mathbf{x}^*)$, this is, for all $\mathbf{x} \in N_\varepsilon(\mathbf{x}^*) \cap \mathcal{F}$, we have that $f(\mathbf{x}) \geq f(\mathbf{x}^*)$ and $\nabla f(\mathbf{x}^*)^T(\mathbf{x} - \mathbf{x}^*) \geq 0$, then we say that $\mathbf{x}^*$ locally solves the test problem.

A.5.3 Test Problem 9

$$\min_{\mathbf{x}\in\mathbb{R}^4} \left[(x_1 + 10x_2)^2 + 5(x_3 - x_4)^2 + (x_2 - 2x_3)^4 + 10(x_1 - x_4)^4\right]$$

subject to

$$C_1: \ \sum_{i=1}^{d} x_i \geq 3$$

$$C_2: \ 2x_1 + \sum_{i=2}^{d} x_i \geq 5$$

Case $d = 4$

At $\mathbf{x}^* = (1.715358,\ -0.132167,\ 0.476726,\ 1.224726)^T$ we have $\mathcal{A}_p(\mathbf{x}^*) = \{2\}$

$$\nabla f(\mathbf{x}^*)^T = (5.5116,\ 2.7559,\ 2.7559,\ 2.7559\ ) \quad \mathbf{a}_2^T = (2,1,1,1)$$

Thus,

$$\begin{bmatrix} 5.5116 \\ 2.7559 \\ 2.7558 \\ 2.7558 \end{bmatrix} - \tilde{u}_2 \begin{bmatrix} 2 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \mathbf{0},$$

for $\tilde{u}_2 = 2.7558$, so $\mathbf{x}^*$satisfied the KT necessary conditions. We can say that the objective function is pseudoconvex for all $\mathbf{x} \in N_\varepsilon(\mathbf{x}^*) \cap \mathcal{F}$, so $\mathbf{x}^*$ is a local minimum of the problem.

Appendix B

Proofs

B.1 Proof of Lemma 4.1

Proof. Let $f(\mathbf{x}) : D \to \mathbb{R}$, where D is a non-empty convex set in $\mathbb{R}^d$. We say that $f(\mathbf{x})$ is strictly convex on D if and only if

$$f(\mu \mathbf{x}_1 + (1-\mu)\mathbf{x}_2) < \mu\, f(\mathbf{x}_1) + (1-\mu)\, f(\mathbf{x}_2) \tag{B.1}$$

for each $\mathbf{x}_1$,$\mathbf{x}_2 \in D$ and for each $0 < \mu < 1$ (Bazaraa and Shetty 1979).

To prove the lemma, we apply the induction method.

Case $k = 2$.

By strictly convexity of $f(\mathbf{x})$, we have

$$f(\mu_1 \mathbf{x}_1 + \mu_2 \mathbf{x}_2) < \mu_1\, f(\mathbf{x}_1) + \mu_2\, f(\mathbf{x}_2) \tag{B.2}$$

for $\{\mathbf{x}_1, \mathbf{x}_2\} \in D$ and for each $0 < \mu_i < 1$, $i = 1, 2$, with $\mu_1 + \mu_2 = 1$.

Case $k = n - 1$.

We assume as true that for any $(n-1) > 2$ points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n-1}\}$ in D, it is satisfied

$$f\left(\sum_{i=1}^{n-1} \mu_i \mathbf{x}_i\right) < \sum_{i=1}^{n-1} \mu_i\, f(\mathbf{x}_i) \tag{B.3}$$

where

$$\sum_{i=1}^{n-1} \mu_i = 1 \text{ and } 0 < \mu_i < 1 \text{ for all } \; i = 1, 2, \ldots, (n-1). \tag{B.4}$$

Case $k = n$.

Now, we prove that for any n points $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_{n-1}, \mathbf{x}_n\} \in D$ the lemma is true.

Using Equations (B.2) and (B.3) to the case $k = n$, we obtain

$$f\left((1-\mu_n)\sum_{i=1}^{n-1} \mu_i \mathbf{x}_i + \mu_n \mathbf{x}_n\right) < (1-\mu_n)\ f\left[\sum_{i=1}^{n-1} \mu_i\ f(\mathbf{x}_i)\right] + \mu_n\ f(\mathbf{x}_n) \qquad \text{(B.5)}$$

$$(1-\mu_n)\ f\left[\sum_{i=1}^{n-1} \mu_i\ f(\mathbf{x}_i)\right] + \mu_n\ f(\mathbf{x}_n) < (1-\mu_n)\sum_{i=1}^{n-1} \mu_i\ f(\mathbf{x}_i) + \mu_n\ f(\mathbf{x}_n) \qquad \text{(B.6)}$$

because $(1-\mu_n)\sum_{i=1}^{n-1} \mu_i + \mu_n = (1-\mu_n) + \mu_n = 1$, as a result of Equation (B.4).

Thus, from Equations (B.5) and (B.6), we have

$$f\left(\sum_{i=1}^{n-1}(1-\mu_n)\mu_i \mathbf{x}_i + \mu_n \mathbf{x}_n\right) < \sum_{i=1}^{n-1}(1-\mu_n)\mu_i\ f(\mathbf{x}_i) + \mu_n\ f(\mathbf{x}_n) \qquad \text{(B.7)}$$

Denoting to $\mathring{\mu}_i = (1-\mu_n)\mu_i$ for all $i = 1, 2, \ldots, (n-1)$, we obtain

$$f\left(\sum_{i=1}^{n-1} \mathring{\mu}_i \mathbf{x}_i + \mu_n \mathbf{x}_n\right) < \sum_{i=1}^{n-1} \mathring{\mu}_i\ f(\mathbf{x}_i) + \mu_n\ f(\mathbf{x}_k),$$

which can be expressed as

$$f\left(\sum_{i=1}^{n} \bar{\mu}_i \mathbf{x}_i\right) < \sum_{i=1}^{n} \bar{\mu}_i\ f(\mathbf{x}_i), \qquad \text{(B.8)}$$

because $\sum_{i=1}^{n-1}(1-\mu_n)\mu_i + \mu_n = \sum_{i=1}^{n-1} \mathring{\mu}_i + \mu_n = 1$.

Note that

$$\bar{\mu}_i = \begin{cases} (1-\mu_n)\mu_i & 1 \le i \le n-1, \\ \mu_n & i = n \end{cases} .$$

Since every strictly convex function is strongly quasiconvex (Bazaraa and Shetty 1979), hence we can assure that

$$f\left(\sum_{i=1}^{n} \mu_i \mathbf{x}_i\right) < \max\{f(\mathbf{x}_1), f(\mathbf{x}_2), \ldots, f(\mathbf{x}_n)\} \qquad \text{(B.9)}$$

■

B.2 Proof of Lemma 4.2

Proof. Let D be a convex set of the d-dimensional Euclidean space $\mathbb{R}^d$. We say that D is convex if, given any two points $\mathbf{x}_1$ and $\mathbf{x}_2$ in D and any scalar μ such that $0 \leq \mu \leq 1$, then the point $\mathbf{x}_\mu = (1-\mu)\mathbf{x}_1 + \mu\mathbf{x}_2$ belongs to D.

To show that $\mathcal{F} \equiv \{\mathbf{x} \in \mathbb{R}^d \mid A\mathbf{x} \geq \mathbf{b}\}$ is convex, we consider that both $\mathbf{x}_1$ and $\mathbf{x}_2$ belong to the feasible set $\mathcal{F}$, so we can affirm that $A\mathbf{x}_1 \geq \mathbf{b}$ and $A\mathbf{x}_2 \geq \mathbf{b}$, and we must verify that the inequality $A\mathbf{x}_\mu \geq \mathbf{b}$ is satisfied, which will be given by

$$A\mathbf{x}_\mu = A\left[(1-\mu)\mathbf{x}_1 + \mu\mathbf{x}_2\right] = (1-\mu)A\mathbf{x}_1 + \mu A\mathbf{x}_2 \geq \mathbf{b} \tag{B.10}$$

because $0 \leq \mu \leq 1$, $A\mathbf{x}_1 \geq \mathbf{b}$ and $A\mathbf{x}_2 \geq \mathbf{b}$. ■

B.3 Proof of Lemma 4.5

Proof. Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two any distinct points belonging to both non-empty convex sets D_1 and D_2. If D_1 is a convex set of $\mathbb{R}^d$, then by defining, the point given by $\mathbf{x}_\mu = (1-\mu)\mathbf{x}_1 + \mu\mathbf{x}_2$ belongs to the set D_1 for all $0 \leq \mu \leq 1$. Thus, we can also affirm that $\mathbf{x}_\mu$ belongs to the set D_2 for all $0 \leq \mu \leq 1$, due to the convexity of the set D_2. Since $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_\mu$ belong to the subset $D_1 \cap D_2$ for all $0 \leq \mu \leq 1$ and for any distinct points $\mathbf{x}_1$ and $\mathbf{x}_2$ in $D_1 \cap D_2$, then we can say that the subset $D_1 \cap D_2$ is a non-empty convex subset. ■

B.4 Proof of Theorem 4.2

Proof. Using Lemma 4.2 we can affirm that the non-empty set $\mathcal{L}$ is convex and, according to Lemma 4.5 we can also say that the subset defined by $\mathcal{L} \cap \Omega$ is a non-empty convex subset, due to the fact that Ω is a non-empty convex set of $\mathbb{R}^d$. Let $\mathbf{x}_1$ and $\mathbf{x}_2$ be two distinct points in the subset $\mathcal{L} \cap \Omega \equiv \{\mathbf{x} \in \mathbb{R}^d \mid a^T\mathbf{x} = b \wedge \mathbf{x} \in \mathbf{\Omega}\}$. Since $\mathbf{x}_1$, $\mathbf{x}_2$ and $\mathbf{x}_\mu = (1-\mu)\mathbf{x}_1 + \mu\mathbf{x}_2$ belong to the subset $\mathcal{L} \cap \Omega$ for all $0 \leq \mu \leq 1$, and $f(\mathbf{x})$ is a strictly convex function on Ω, then $f(\mathbf{x})$ is a strictly convex function on $\mathcal{L} \cap \Omega$. ∎

B.5 Proof of Corollary 4.1

Proof. Let $\mathcal{E}_i$ be a non-empty, non-singleton and convex subset of $\mathbb{R}^d$ defined by a linear equality such that $\mathcal{E}_i \equiv \{\mathbf{x} \in \mathbb{R}^d \mid a_i^T\mathbf{x} = b_i \}$, where the matrix $A \in \mathbb{R}^{l \times d}$ is a constant matrix such that $A = \left[a_1^T : \cdots : a_l^T\right]^T$ and, the vector $\mathbf{b} \in \mathbb{R}^d$ is formed by $\mathbf{b} = (b_1, \ldots, b_l)^T$. Thus, we can say that $\mathcal{E} = \bigcap_{i=1}^{l} \mathcal{E}_i$, which is assumed as non-empty subset of $\mathbb{R}^d$.

On the other hand, let $\mathcal{E}_{1,k}$ denote a subset defined by $\mathcal{E}_{1,k} = \bigcap_{i=1}^{k} \mathcal{E}_i$, where k is a positive integer number. Assume that $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ is a strictly convex function on a non-empty convex set Ω of $\mathbb{R}^d$ and, $\mathcal{E} \cap \Omega$ as a non-empty convex subset of Ω.

Using Lemma 4.2 we can say that $f(\mathbf{x})$ is a strictly convex function on a non-empty and convex subset $\mathcal{E}_{1,2} = \mathcal{E}_1 \cap \mathcal{E}_2$. Thus, we can affirm that $f(\mathbf{x})$ is a strictly convex function on a non-empty and convex subset $\mathcal{E}_{1,3} = \mathcal{E}_{1,2} \cap \mathcal{E}_3 = \mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3$. Using the same argument, we say that $f(\mathbf{x})$ is a strictly convex function on a non-empty and convex subset $\mathcal{E}_{1,l-1} = \bigcap_{i=1}^{l-1} \mathcal{E}_i$. If we add the last linear equality to the proof, we have that $f(\mathbf{x})$ is a strictly convex function on a non-empty and convex subset $\mathcal{E} = \mathcal{E}_{1,l-1} \cap \mathcal{E}_l = \bigcap_{i=1}^{l} \mathcal{E}_i$ due to Lemma 4.2. Finally, applying Lemma 4.2 to the non-empty convex subset $\mathcal{E} \cap \Omega$ of Ω, we prove that $f(\mathbf{x})$ is a strictly convex function on a non-empty and convex subset $\mathcal{E} \cap \Omega$ of Ω. ∎

Appendix C

Constrained linear objective function

C.1 The value of μ_0

Since $\mathbf{x}_{new}^0$ and $\mathbf{x}_2^0$ are collinear, and $\mathbf{x}_3^0$ and $\mathbf{x}_1^0$ are collinear, we have

$$\mathbf{x}_{new}^0 = \mu_0 \mathbf{x}_2^0 \text{ and } \mathbf{x}_3^0 = k_1 k_2 \ \mathbf{x}_1^0 \tag{C.1}$$

On the other hand,

$$\mathbf{x}_{new}^0 = \mathbf{x}_3^0 + \nu \left[\frac{1}{2}(\mathbf{x}_1^0 + \mathbf{x}_2^0) - \mathbf{x}_3^0 \right] \tag{C.2}$$

Using Equations (C.1) and (C.2), we obtain

$$\mu_0 \mathbf{x}_2^0 = k_1 k_2 \ \mathbf{x}_1^0 + \frac{\nu}{2}\mathbf{x}_1^0 + \frac{\nu}{2}\mathbf{x}_2^0 - \nu k_1 k_2 \ \mathbf{x}_1^0,$$

Therefore

$$(\mu_0 - \frac{\nu}{2})\mathbf{x}_2^0 = (k_1 k_2 \ + \frac{\nu}{2} - \nu k_1 k_2)\mathbf{x}_1^0 \tag{C.3}$$

Equation (C.3) is satisfied if and only if

$$\mu_0 - \tfrac{\nu}{2} = 0$$
$$k_1 k_2 + \tfrac{\nu}{2} - \nu k_1 k_2 = 0$$

so

$$\nu = 2\mu_0 \text{ and } k_1 k_2 + \mu_0 - 2\mu_0 k_1 k_2 = 0$$

which is obtained,

$$\mu_0 = \frac{k_1 k_2}{2k_1 k_2 - 1} = \frac{1}{2 - \frac{1}{k_1 k_2}}$$

C.2 Computing $g_o(n)$ and $g_e(n)$

Since, the terms in Equations (4.65) and (4.70) are the same, we shall compute

$$p_1(n) = \prod_{i=1}^{n-1} \left[1 + \frac{1/2w}{i}\right]^2 \quad \forall n = 1, 2, \ldots, \tag{C.4}$$

$$p_2(n) = \prod_{i=1}^{n-1} \left[1 + \frac{(w+1)/2w}{i}\right]^2 \quad \forall n = 1, 2, \ldots, \tag{C.5}$$

and

$$p_3(n) = \prod_{i=1}^{n-1} \left[1 + \frac{(1-w)/2w}{i}\right]^2 \quad \forall 0 < w < 1 \ \wedge \ n = 1, 2, \ldots . \tag{C.6}$$

Using the variant of the gamma function $\Gamma_m(z)$ (Olver 1974; Markushevich 1965), we have

$$\frac{1}{\Gamma_m(z)} = z \exp(z\gamma_m) \prod_{i=1}^{m} \left[\left(1 + \frac{z}{i}\right) \exp(-z/i)\right], \tag{C.7}$$

where $\gamma_m = \sum_{i=1}^{m} \frac{1}{i} - \ln m$ and here m is any positive integer number.

Letting $m \to \infty$, we obtain the well-known constant of Euler γ, that is

$$\lim_{m\to\infty}\gamma_m = \lim_{m\to\infty}\left[\sum_{i=1}^{m}\frac{1}{i} - \ln(m)\right] = \gamma.$$

Therefore Equation (C.7) becomes

$$\lim_{m\to\infty}\frac{1}{\Gamma_m(z)} = \frac{1}{\Gamma(z)} = z\exp(z\gamma)\prod_{i=1}^{\infty}\left[\left(1+\frac{z}{i}\right)\exp(-z/i)\right]$$

Calculating the natural logarithm to Equation (C.7), we obtain

$$\sum_{i=1}^{m}\ln\left(1+\frac{z}{i}\right)^2 = 2z\ln m - \ln(z^2) - \ln\Gamma_m^2(z), \tag{C.8}$$

Equation (C.4) can be transformed by the natural logarithm, thus

$$S_1(n) = \sum_{i=1}^{n-1}\ln\left[1+\frac{1/2w}{i}\right]^2 \quad \forall n = 1,2,\dots,$$

which can be rewritten

$$S_1(n) = 2\frac{1}{2w}\ln(n-1) - \ln\left(\frac{1}{(2w)^2}\right) - \ln\Gamma_{n-1}^2\left(\frac{1}{2w}\right) \quad \forall n = 1,2,\dots. \tag{C.9}$$

Applying the same method to Equation (C.5), we have

$$S_2(n) = \sum_{i=1}^{n-1}\ln\left[1+\frac{(w+1)/2w}{i}\right]^2 \quad \forall n = 1,2,\dots,$$

so,

$$S_2(n) = 2\frac{(w+1)}{2w}\ln(n-1) - \ln\left(\left(\frac{w+1}{2w}\right)^2\right) - \ln\Gamma_{n-1}^2\left(\frac{w+1}{2w}\right) \quad \forall n = 1,2,\dots. \tag{C.10}$$

Using the same procedure to Equation (C.6), we obtain

$$S_3(n) = \sum_{i=1}^{n-1}\ln\left[1+\frac{(1-w)/2w}{i}\right]^2 \quad \forall 0 < w < 1 \;\wedge\; n = 1,2,\dots,$$

therefore for all $0 < w < 1$ and $n = 1, 2, \dots$,

$$S_3(n) = 2\frac{(1-w)}{2w}\ln(n-1) - \ln\left(\left(\frac{1-w}{2w}\right)^2\right) - \ln\Gamma^2_{n-1}\left(\frac{1-w}{2w}\right) \tag{C.11}$$

Using Equations (C.9) and (C.10), in $\ln[g_o(n)]$, we have

$$\ln[g_o(n)] = S_1(n) - S_2(n)$$

$$g_o(n) = \frac{(w+1)^2}{(n-1)}\frac{\Gamma^2_{n-1}\left(\frac{w+1}{2w}\right)}{\Gamma^2_{n-1}\left(\frac{1}{2w}\right)} \quad \forall n = 1, 2, \dots . \tag{C.12}$$

For calculating $g_e(n)$ in terms of $\Gamma_d(z)$, we have

$$\ln[g_e(n)] = S_3(n) - S_1(n)$$

so,

$$g_e(n) = \frac{1}{(n-1)(1-w)^2}\frac{\Gamma^2_{n-1}\left(\frac{1}{2w}\right)}{\Gamma^2_{n-1}\left(\frac{1-w}{2w}\right)} \quad \forall 0 < w < 1 \quad \wedge \quad n = 1, 2, \dots . \tag{C.13}$$

Appendix D

Reports of experiments

D.1 Reports of experiments: obtuse solid angle of feasible cone

Table D.1: Summary of test problem TP1(2).

Exp	PM	NE	DTP	F
1	582	582	0	1
2	770	770	0	1
3	406	406	0	1
4	388	388	0	1
5	399	399	0	1
6	5000	1000	0	0
7	4233.1	231	0.21	0
8	477	477	0	1
9	644	644	0	1
10	24	24	0	1
11	23	23	0	1
12	22	22	0	1
13	60	60	0	1
14	61	61	0	1
15	26	26	0	1
16	27	27	0	1
17	66	66	0	1
18	61	61	0	1
19	25	25	0	1
20	23	23	0	1
21	24	24	0	1
22	34	34	0	1
23	38	38	0	1
24	27	27	0	1
25	37	37	0	1
26	41	41	0	1
27	26	26	0	1

Table D.2: Summary of test problem TP1(4).

Exp	PM	NE	DTP	F
1	5006	1000+	0.6	0
2	1002.2	1000+	0.22	1
3	5017.3	1000+	1.73	0
4	5002.8	1000+	0.28	0
5	1000.18	1000+	0.018	1
6	1003.34	1000+	0.334	1
7	1000.66	1000+	0.066	1
8	5009.88	1000+	0.988	0
9	5026.7	1000+	2.67	0
10	389	389	0	1
11	473	473	0	1
12	419	419	0	1
13	373	373	0	1
14	364	364	0	1
15	425	425	0	1
16	388	388	0	1
17	365	365	0	1
18	444	444	0	1
19	277	277	0	1
20	325	325	0	1
21	330	330	0	1
22	292	292	0	1
23	267	267	0	1
24	407	407	0	1
25	547	547	0	1
26	293	293	0	1
27	324	324	0	1

[0] The symbol "+" represents that the method was stopped at the indicated number of evaluation.

Table D.3: Summary of test problem TP1(6).

Exp	PM	NE	DTP	F
1	20006.3	20000+	0.63	1
2	20006.7	20000+	0.67	1
3	24027.4	20000+	2.74	0
4	20007.6	20000+	0.76	1
5	20008.9	20000+	0.89	1
6	20008.3	20000+	0.83	1
7	20009.2	20000+	0.92	1
8	24019	20000+	1.9	0
9	20016.4	20000+	1.64	1
10	635	635	0	1
11	1634	1634	0	1
12	1746	1746	0	1
13	18484.4	18473	1.14	1
14	850	850	0	1
15	1321	1321	0	1
16	2433	2433	0	1
17	1115	1115	0	1
18	823	823	0	1
19	644	644	0	1
20	1315	1315	0	1
21	1199	1199	0	1
22	20015.7	20000+	1.57	1
23	1261	1261	0	1
24	1034	1034	0	1
25	576	576	0	1
26	1327	1327	0	1
27	596	596	0	1

Table D.4: Summary of test problem TP2(2).

Exp	PM	NE	DTP	F
1	3459	3459	0	1
2	3515	3515	0	1
3	147	147	0	1
4	3525	3525	0	1
5	3492	3492	0	1
6	3461	3461	0	1
7	3518	3518	0	1
8	3608	3608	0	1
9	4111.8	4000+	11.18	1
10	566	566	0	1
11	3710	3710	0	1
12	1963	1963	0	1
13	126.2	105	2.12	1
14	126.2	105	2.12	1
15	136.2	115	2.12	1
16	127.2	106	2.12	1
17	129.2	108	2.12	1
18	352.2	331	2.12	1
19	726	726	0	1
20	471	471	0	1
21	765	765	0	1
22	905	905	0	1
23	924	924	0	1
24	500	500	0	1
25	528	528	0	1
26	548	548	0	1
27	2575	2575	0	1

Table D.5: Summary of test problem TP2(4).

Exp	PM	NE	DTP	F
1	12000	12000+	0	1
2	12000	12000+	0	1
3	12000	12000+	0	1
4	12000	12000+	0	1
5	12000	12000+	0	1
6	12000	12000+	0	1
7	12000	12000+	0	1
8	12000	12000+	0	1
9	12000	12000+	0	1
10	1860	1860	0	1
11	12000	12000	0	1
12	9093	9093	0	1
13	1531	1531	0	1
14	1364	1364	0	1
15	3773	3773	0	1
16	3286	3286	0	1
17	404	389	1.5	1
18	2413	2413	0	1
19	2447	2447	0	1
20	1009	1009	0	1
21	11031	11031	0	1
22	8305	8305	0	1
23	1071	1071	0	1
24	750	750	0	1
25	2935	2935	0	1
26	6583	6583	0	1
27	1637	1637	0	1

Table D.6: Summary of test problem TP2(6).

Exp	PM	NE	DTP	F
1	12000	12000+	0	1
2	12000	12000+	0	1
3	12062.8	12000+	6.28	1
4	12000	12000+	0	1
5	12000	12000+	0	1
6	12000	12000+	0	1
7	12000	12000+	0	1
8	12000	12000+	0	1
9	12000	12000+	0	1
10	3129	3129	0	1
11	4799	4799	0	1
12	3494	3494	0	1
13	9442	9442	0	1
14	4034	4034	0	1
15	9255	9255	0	1
16	2198	2198	0	1
17	10971	10971	0	1
18	3092	3092	0	1
19	1644	1644	0	1
20	3105	3105	0	1
21	3332	3332	0	1
22	9449	9449	0	1
23	7651	7651	0	1
24	2180	2180	0	1
25	7597	7597	0	1
26	6452	6452	0	1
27	3532	3532	0	1

Table D.7: Summary of test problem TP3(2).

Exp	PM	NE	DTP	F
1	727	727	0	1
2	543	543	0	1
3	327.8	325	0.28	1
4	409	409	0	1
5	348	348	0	1
6	1080	1000+	8	1
7	233	231	0.2	1
8	464	464	0	1
9	711	711	0	1
10	25	25	0	1
11	22	22	0	1
12	23	23	0	1
13	59	59	0	1
14	62	62	0	1
15	26	26	0	1
16	60	60	0	1
17	32	32	0	1
18	61	61	0	1
19	25	25	0	1
20	23	23	0	1
21	23	23	0	1
22	33	33	0	1
23	38	38	0	1
24	27	27	0	1
25	27	27	0	1
26	32	32	0	1
27	26	26	0	1

Table D.8: Summary of test problem TP3(4).

Exp	PM	NE	DTP	F
1	1005	1000+	0.5	1
2	1002.2	1000+	0.22	1
3	5039.7	1000+	3.97	0
4	5031	1000+	3.1	0
5	5013.3	1000+	1.33	0
6	5008.4	1000+	0.84	0
7	5009.4	1000+	0.94	0
8	5019.2	1000+	1.92	0
9	1001.2	1000+	0.12	1
10	413	413	0	1
11	395	395	0	1
12	376	376	0	1
13	417	417	0	1
14	357	357	0	1
15	269	269	0	1
16	417	417	0	1
17	381	381	0	1
18	420	420	0	1
19	600	600	0	1
20	267	267	0	1
21	270	270	0	1
22	486	486	0	1
23	316	316	0	1
24	312	312	0	1
25	351	351	0	1
26	283	283	0	1
27	386	386	0	1

Table D.9: Summary of test problem TP3(6).

Exp	PM	NE	DTP	F
1	2000	2000+	0	1
2	2018.8	2000+	1.88	1
3	2010.6	2000+	1.06	1
4	2010.5	2000+	1.05	1
5	2011.1	2000+	1.11	1
6	2018	2000+	1.8	1
7	2010.6	2000+	1.06	1
8	2012.3	2000+	1.23	1
9	2008	2000+	0.8	1
10	796	796	0	1
11	1730	1730	0	1
12	720	720	0	1
13	1206	1206	0	1
14	1238	1238	0	1
15	1192	1192	0	1
16	1069	1069	0	1
17	746	746	0	1
18	1806	1806	0	1
19	568	568	0	1
20	1854	1854	0	1
21	480	480	0	1
22	900	900	0	1
23	563	563	0	1
24	695	695	0	1
25	641	641	0	1
26	603	603	0	1
27	642	642	0	1

Table D.10: Summary of test problem TP4(2).

Exp	PM	NE	DTP	F
1	267	267	0	1
2	282	282	0	1
3	147	147	0	1
4	285	285	0	1
5	274	274	0	1
6	272	272	0	1
7	287	287	0	1
8	293	293	0	1
9	1111.8	1000+	11.18	1
10	426	426	0	1
11	729	729	0	1
12	1022.3	1000	2.23	1
13	126.2	105	2.12	1
14	126.2	105	2.12	1
15	136.2	115	2.12	1
16	135.2	114	2.12	1
17	185	185	0	1
18	1000	1000+	0	1
19	226	226	0	1
20	234	234	0	1
21	1000	1000	0	1
22	285	285	0	1
23	192	192	0	1
24	1000	1000+	0	1
25	212	212	0	1
26	203	203	0	1
27	1000	1000+	0	1

Table D.11: Summary of test problem TP4(4).

Exp	PM	NE	DTP	F
1	536	536	0	1
2	580	580	0	1
3	700	700	0	1
4	549	549	0	1
5	556	556	0	1
6	723	723	0	1
7	576	576	0	1
8	635	635	0	1
9	577	577	0	1
10	1000	1000+	0	1
11	1000	1000+	0	1
12	1000	1000+	0	1
13	1000	1000+	0	1
14	839	839	0	1
15	1000	1000+	0	1
16	780	780	0	1
17	1000	1000+	0	1
18	412	397	1.5	1
19	775	775	0	1
20	501	501	0	1
21	1000	1000+	0	1
22	1000	1000+	0	1
23	1000	1000+	0	1
24	1000	1000+	0	1
25	1000	1000+	0	1
26	1000	1000+	0	1
27	318	303	1.5	1

Table D.12: Summary of test problem TP4(6).

Exp	PM	NE	DTP	F
1	872	872	0	1
2	1090	1090	0	1
3	1356	1356	0	1
4	893	893	0	1
5	1230	1230	0	1
6	1081	1081	0	1
7	947	947	0	1
8	937	937	0	1
9	1294	1294	0	1
10	807	807	0	1
11	760	760	0	1
12	2000	2000	0	1
13	739	739	0	1
14	2000	2000+	0	1
15	2000	2000+	0	1
16	2000	2000+	0	1
17	643	643	0	1
18	2000	2000+	0	1
19	772	772	0	1
20	2000	2000+	0	1
21	2000	2000+	0	1
22	2000	2000+	0	1
23	2000	2000+	0	1
24	2000	2000+	0	1
25	2000	2000+	0	1
26	789	789	0	1
27	2000	2000+	0	1

Table D.13: Summary of test problem TP5(2).

Exp	PM	NE	DTP	F
1	426	426	0	1
2	537	537	0	1
3	272	272	0	1
4	274	274	0	1
5	549	549	0	1
6	690	690	0	1
7	311.34	207	10.434	1
8	483	483	0	1
9	552.5	443	10.95	1
10	234.5	125	10.95	1
11	216.5	107	10.95	1
12	226.5	117	10.95	1
13	288.5	179	10.95	1
14	256.5	147	10.95	1
15	235.5	126	10.95	1
16	198	198	0	1
17	202.5	93	10.95	1
18	238.5	129	10.95	1
19	201.5	92	10.95	1
20	245.5	136	10.95	1
21	220.5	111	10.95	1
22	267.5	158	10.95	1
23	246.5	137	10.95	1
24	227.5	118	10.95	1
25	182	182	0	1
26	198.5	89	10.95	1
27	229.5	120	10.95	1

Table D.14: Summary of test problem TP5(4).

Exp	PM	NE	DTP	F
1	1086.6	850	23.66	1
2	901.7	796	10.57	1
3	1039.1	830	20.91	1
4	3706	3706	0	1
5	1664.4	1484	18.04	1
6	1656.5	1654	0.25	1
7	26000.04	26000+	0.004	1
8	1779.7	1567	21.27	1
9	2065	1711	35.4	1
10	1233	1233	0	1
11	1682	1682	0	1
12	1402	1402	0	1
13	25742.85	25733	0.985	1
14	1601	1601	0	1
15	2129	2129	0	1
16	7723.47	7687	3.647	1
17	1372	1372	0	1
18	1535	1535	0	1
19	1454	1454	0	1
20	1298	1298	0	1
21	1244	1244	0	1
22	11388.6	11387	0.1595	1
23	1120	1120	0	1
24	1396	1396	0	1
25	1344	1344	0	1
26	1329	1329	0	1
27	1212	1212	0	1

Table D.15: Summary of test problem TP5(6).

Exp	PM	NE	DTP	F
1	7321.4	3215	10.64	0
2	5484.8	1241	24.38	0
3	1607.2	1356	25.12	1
4	2028.729	2020	0.8729	1
5	3407.288	3268	13.9288	1
6	1486.985	1348	13.8985	1
7	19889.51	19861	2.8514	1
8	4968.683	4650	31.86827	1
9	1468.797	1176	29.27969	1
10	5298.187	5028	27.0187	1
11	2888.583	2813	7.5583	1
12	2554.762	2483	7.17616	1
13	6124.045	6068	5.6045	1
14	5547.543	5332	21.5543	1
15	2911.174	2656	25.51743	1
16	2505.177	2331	17.41772	1
17	6008.028	6008	0.00275	1
18	2737.278	2535	20.2278	1
19	1922.953	1894	2.8953	1
20	3294.939	3291	0.39388	1
21	3736.453	3715	2.1453	1
22	3378.252	3279	9.92518	1
23	2391.457	2335	5.64568	1
24	2912.269	2781	13.12693	1
25	19878.87	19861	1.787067	1
26	2657.926	2560	9.792611	1
27	3104	3104	0	1

Table D.16: Summary of test problem TP6(2).

Exp	PM	NE	DTP	F
1	398	398	0	1
2	389	389	0	1
3	302	302	0	1
4	206	206	0	1
5	404	404	0	1
6	409	409	0	1
7	204	204	0	1
8	280	280	0	1
9	463	463	0	1
10	350	350	0	1
11	318	318	0	1
12	305	305	0	1
13	378	378	0	1
14	363	363	0	1
15	338	338	0	1
16	175	175	0	1
17	336	336	0	1
18	303	303	0	1
19	330	330	0	1
20	350	350	0	1
21	334	334	0	1
22	351	351	0	1
23	352	352	0	1
24	338	338	0	1
25	218	218	0	1
26	365	365	0	1
27	331	331	0	1

Table D.17: Summary of test problem TP6(4).

Exp	PM	NE	DTP	F
1	1086.6	850	23.66	1
2	901.7	796	10.57	1
3	1009.1	830	17.91	1
4	3218	3218	0	1
5	1664.4	1484	18.04	1
6	1656.5	1654	0.25	1
7	1695	1695	0	1
8	1779.7	1567	21.27	1
9	2065.1	1711	35.41	1
10	1218	1218	0	1
11	1558	1558	0	1
12	1782	1782	0	1
13	1145	1145	0	1
14	1567	1567	0	1
15	2272	2272	0	1
16	2494	2494	0	1
17	1851	1851	0	1
18	1885	1885	0	1
19	1368	1368	0	1
20	1098	1098	0	1
21	1260	1260	0	1
22	1006	1006	0	1
23	1172	1172	0	1
24	1667	1667	0	1
25	1288	1288	0	1
26	1649	1649	0	1
27	1025	1025	0	1

Table D.18: Summary of test problem TP6(6).

Exp	PM	NE	DTP	F
1	3321.4	3215	10.64	1
2	1484.8	1241	24.38	1
3	1607.2	1356	25.12	1
4	2477.07	2469	0.807	1
5	3407.29	3268	13.929	1
6	1486.99	1348	13.899	1
7	1731.8	1714	1.78	1
8	4968.6	4650	31.86	1
9	1468.8	1176	29.28	1
10	5298.2	5028	27.02	1
11	2888.6	2813	7.56	1
12	2554.76	2483	7.176	1
13	6124.05	6068	5.605	1
14	5547.54	5332	21.554	1
15	2911.2	2656	25.52	1
16	2505.17	2331	17.417	1
17	4798.66	4778	2.066	1
18	2737.28	2535	20.228	1
19	7777	7777	0	1
20	3294.9	3291	0.39	1
21	2676.07	2666	1.007	1
22	3378.3	3279	9.93	1
23	2638.11	2584	5.411	1
24	2912.27	2781	13.127	1
25	2969.57	2954	1.557	1
26	2657.93	2560	9.793	1
27	2891.57	2888	0.357	1

Table D.19: Summary of test problem TP7(4).

Exp	PM	NE	DTP	F
1	6023.6	6000+	2.36	1
2	6000	6000+	0	1
3	5752.2	5747	0.52	1
4	650	650	0	1
5	6000	6000+	0	1
6	10018.3	6000+	1.83	0
7	6023.95	6000+	2.395	1
8	6016.3	6000+	1.63	1
9	10027.4	6000+	2.74	0
10	577	577	0	1
11	637	637	0	1
12	527	527	0	1
13	687.6	664	2.36	1
14	554	554	0	1
15	839.6	816	2.36	1
16	444	444	0	1
17	514.6	491	2.36	1
18	514.6	491	2.36	1
19	441	441	0	1
20	372	372	0	1
21	377	377	0	1
22	663	663	0	1
23	397	397	0	1
24	461.6	438	2.36	1
25	392	392	0	1
26	424.6	401	2.36	1
27	498.6	475	2.36	1

Table D.20: Summary of test problem TP8(4).

Exp	PM	NE	DTP	F
1	756.69	737	1.969	1
2	646.69	627	1.969	1
3	736	736	0	1
4	534	534	0	1
5	524	524	0	1
6	580.69	561	1.969	1
7	571.69	552	1.969	1
8	809	809	0	1
9	819.69	800	1.969	1
10	389.69	370	1.969	1
11	396	396	0	1
12	378	378	0	1
13	579.69	560	1.969	1
14	394	394	0	1
15	414.69	395	1.969	1
16	346	346	0	1
17	496.69	477	1.969	1
18	444.69	425	1.969	1
19	459.69	440	1.969	1
20	422	422	0	1
21	460	460	0	1
22	475	475	0	1
23	438	438	0	1
24	546.69	527	1.969	1
25	403	403	0	1
26	580.69	561	1.969	1
27	476.69	457	1.969	1

Table D.21: Summary of test problem TP9(4).

Exp	PM	NE	DTP	F
1	2000.75	2000+	0.075	1
2	2007.11	2000+	0.711	1
3	2001.6	2000+	0.16	1
4	2005	2000+	0.5	1
5	6016.1	2000+	1.61	0
6	6007.1	2000+	0.71	0
7	1386.2	1386	0.02	1
8	6007.2	2000+	0.72	0
9	2010.7	2000+	1.07	1
10	455	455	0	1
11	462	462	0	1
12	521	521	0	1
13	263	263	0	1
14	450	450	0	1
15	655	655	0	1
16	529	529	0	1
17	646	646	0	1
18	418	418	0	1
19	289	289	0	1
20	377	377	0	1
21	343	343	0	1
22	290	290	0	1
23	549	549	0	1
24	313	313	0	1
25	568	568	0	1
26	367	367	0	1
27	342	342	0	1

Table D.22: Summary of test problem TP10(4).

Exp	PM	NE	DTP	F
1	3381	3381	0	1
2	2636	2636	0	1
3	3395	3395	0	1
4	3125	3125	0	1
5	2754	2754	0	1
6	3134	3134	0	1
7	2798	2798	0	1
8	3396	3396	0	1
9	2874	2874	0	1
10	1199	1199	0	1
11	1629	1629	0	1
12	2423	2423	0	1
13	1799	1799	0	1
14	1756	1756	0	1
15	2174	2174	0	1
16	1741	1741	0	1
17	2095	2095	0	1
18	373.38	350	2.338	1
19	1809	1809	0	1
20	2050	2050	0	1
21	2640	2640	0	1
22	1775	1775	0	1
23	1316	1316	0	1
24	1560	1560	0	1
25	1270	1270	0	1
26	1627	1627	0	1
27	1718	1718	0	1

D.2 Reports of experiments: non-obtuse solid angle of feasible cone

Table D.23: Summary of test problem TP11(2).

Exp	PM	NE	DTP	F
1	289	289	0	1
2	4290	290	0	0
3	4357	357	0	0
4	4313	313	0	0
5	4313	313	0	0
6	400	400+	0	1
7	4313	313	0	0
8	4313	313	0	0
9	400	400+	0	1
10	22	22	0	1
11	22	22	0	1
12	21	21	0	1
13	81.75	61	2.075	1
14	39	39	0	1
15	101.9	51	5.09	1
16	81.75	61	2.075	1
17	39	39	0	1
18	101.5	51	5.05	1
19	24	24	0	1
20	22	22	0	1
21	22	22	0	1
22	31	31	0	1
23	32	32	0	1
24	29	29	0	1
25	31	31	0	1
26	32	32	0	1
27	29	29	0	1

Table D.24: Summary of test problem TP11(4).

Exp	PM	NE	DTP	F
1	5500.3	1500+	0.03	0
2	5509.1	1500+	0.91	0
3	5536.9	1500+	3.69	0
4	5502.6	1500+	0.26	0
5	5502.6	1500+	0.26	0
6	5532.1	1500+	3.21	0
7	5501.71	1500+	0.171	0
8	5520.8	1500+	2.08	0
9	5511.9	1500+	1.19	0
10	301	301	0	1
11	483	483	0	1
12	254	254	0	1
13	332	332	0	1
14	281	281	0	1
15	359	359	0	1
16	255	255	0	1
17	255	255	0	1
18	344	344	0	1
19	504	504	0	1
20	326	326	0	1
21	231	231	0	1
22	1329	1329	0	1
23	1500	1500+	0	1
24	437	437	0	1
25	371	371	0	1
26	345	345	0	1
27	344	344	0	1

Table D.25: Summary of test problem TP11(6).

Exp	PM	NE	DTP	F
1	9505.46	5500+	0.546	0
2	9523.58	5500+	2.358	0
3	9513.91	5500+	1.391	0
4	9506.6	5500+	0.66	0
5	9521.9	5500+	2.19	0
6	9527.8	5500+	2.78	0
7	9503.69	5500+	0.369	0
8	9505.88	5500+	0.588	0
9	9537.16	5500+	3.716	0
10	1205	1205	0	1
11	1556	1556	0	1
12	2068	2068	0	1
13	1229	1229	0	1
14	921	921	0	1
15	1138	1138	0	1
16	3842	3842	0	1
17	698	698	0	1
18	5443	5443	0	1
19	4776	4776	0	1
20	1529	1529	0	1
21	707	707	0	1
22	1240	1240	0	1
23	1878	1878	0	1
24	1419	1419	0	1
25	1243	1243	0	1
26	2516	2516	0	1
27	1999	1999	0	1

Table D.26: Summary of test problem TP12(2).

Exp	PM	NE	DTP	F
1	3573	3573	0	1
2	3483	3483	0	1
3	3645	3600	4.5	1
4	3500	3500	0	1
5	3567	3567	0	1
6	3600	3600	0	1
7	3564	3564	0	1
8	3575	3575	0	1
9	3628.2	3600	2.82	1
10	1479	1479	0	1
11	2190	2190	0	1
12	122.9	114	0.89	1
13	124.9	116	0.89	1
14	129.9	121	0.89	1
15	125.9	117	0.89	1
16	127.9	119	0.89	1
17	164.9	156	0.89	1
18	162.9	154	0.89	1
19	518	518	0	1
20	3077	3077	0	1
21	658	658	0	1
22	2887	2887	0	1
23	3260	3260	0	1
24	107.9	99	0.89	1
25	109.9	101	0.89	1
26	345	345	0	1
27	96	96	0	1

Table D.27: Summary of test problem TP12(4).

Exp	PM	NE	DTP	F
1	8514	8514	0	1
2	9670	9670	0	1
3	9689	9689	0	1
4	9873	9873	0	1
5	9207	9207	0	1
6	9897	9897	0	1
7	8629	8629	0	1
8	9468	9468	0	1
9	9826	9826	0	1
10	4427	4427	0	1
11	936	936	0	1
12	6334	6334	0	1
13	2476	2476	0	1
14	2722	2722	0	1
15	9373	9373	0	1
16	1820	1820	0	1
17	7577	7577	0	1
18	847	847	0	1
19	2535	2535	0	1
20	634	634	0	1
21	2350	2350	0	1
22	1546	1546	0	1
23	1021	1021	0	1
24	978	978	0	1
25	1012	1012	0	1
26	1843	1843	0	1
27	1798	1798	0	1

Table D.28: Summary of test problem TP12(6).

Exp	PM	NE	DTP	F
1	17255	17255	0	1
2	19539	19539	0	1
3	24125	24125	0	1
4	17123	17123	0	1
5	20036	20036	0	1
6	23817	23817	0	1
7	17107	17107	0	1
8	19107	19107	0	1
9	25519	25519	0	1
10	2579	2579	0	1
11	3577	3577	0	1
12	2240	2240	0	1
13	16437	16437	0	1
14	1394	1394	0	1
15	3707	3707	0	1
16	6184	6184	0	1
17	3097	3097	0	1
18	3713	3713	0	1
19	13248	13248	0	1
20	1722	1722	0	1
21	1745	1745	0	1
22	4623	4623	0	1
23	3729	3729	0	1
24	3091	3091	0	1
25	7169	7169	0	1
26	1850	1850	0	1
27	1739	1739	0	1

Table D.29: Summary of test problem TP13(2).

Exp	PM	NE	DTP	F
1	290	290	0	1
2	290	290	0	1
3	357	357	0	1
4	314	314	0	1
5	314	314	0	1
6	400	400+	0	1
7	314	314	0	1
8	314	314	0	1
9	400	400+	0	1
10	23	23	0	1
11	22	22	0	1
12	20	20	0	1
13	81.7	61	2.07	1
14	39	39	0	1
15	102.4	52	5.04	1
16	81.8	61	2.08	1
17	38	38	0	1
18	102.4	52	5.04	1
19	24	24	0	1
20	23	23	0	1
21	23	23	0	1
22	32	32	0	1
23	32	32	0	1
24	30	30	0	1
25	32	32	0	1
26	32	32	0	1
27	31	31	0	1

Table D.30: Summary of test problem TP13(4).

Exp	PM	NE	DTP	F
1	4701.87	700+	0.187	0
2	4702.74	700+	0.274	0
3	4737.6	700+	3.76	0
4	700	700+	0	1
5	4702.3	700+	0.23	0
6	4798.6	700+	9.86	0
7	700	700+	0	1
8	4767.4	700+	6.74	0
9	4715.4	700+	1.54	0
10	263	263	0	1
11	195	195	0	1
12	361	361	0	1
13	249	249	0	1
14	200	200	0	1
15	365	365	0	1
16	327	327	0	1
17	252	252	0	1
18	307	307	0	1
19	304	304	0	1
20	267	267	0	1
21	444	444	0	1
22	700	700	0	1
23	608	608	0	1
24	465	465	0	1
25	401	401	0	1
26	400	400	0	1
27	381	381	0	1

Table D.31: Summary of test problem TP13(6).

Exp	PM	NE	DTP	F
1	13102.7	9100+	0.27	0
2	13224.4	9100+	12.44	0
3	13588.3	9100+	48.83	0
4	13103.78	9100+	0.378	0
5	19820	9100+	672	0
6	13190.4	9100+	9.04	0
7	5337.12	569	76.812	0
8	5344.12	576	76.812	0
9	5642.12	874	76.812	0
10	570.06	569	0.106	1
11	576	576	0	1
12	874	874	0	1
13	9017.06	9016	0.106	1
14	1555	1555	0	1
15	4892	4892	0	1
16	1926	1926	0	1
17	1672	1672	0	1
18	9109.43	9100+	0.943	1
19	1036	1036	0	1
20	9100	9100+	0	1
21	668.42	668	0.042	1
22	5992.43	5991	0.143	1
23	2553	2553	0	1
24	1390	1390	0	1
25	9100.528	9100+	0.0528	1
26	1710	1710	0	1
27	1288	1288	0	1

Table D.32: Summary of test problem TP14(2).

Exp	PM	NE	DTP	F
1	307	307	0	1
2	278	278	0	1
3	445	400+	4.5	1
4	290	290	0	1
5	279	279	0	1
6	400	400+	0	1
7	310	310	0	1
8	321	321	0	1
9	428.2	400+	2.82	1
10	400	400+	0	1
11	428.2	400+	2.82	1
12	123.9	115	0.89	1
13	124.9	116	0.89	1
14	123.9	115	0.89	1
15	128.9	120	0.89	1
16	127.9	119	0.89	1
17	164.9	156	0.89	1
18	135.9	127	0.89	1
19	317	317	0	1
20	310	310	0	1
21	105.9	97	0.89	1
22	306	306	0	1
23	363	363	0	1
24	105.9	97	0.89	1
25	108.9	100	0.89	1
26	408.9	400+	0.89	1
27	103.9	95	0.89	1

Table D.33: Summary of test problem TP14(4).

Exp	PM	NE	DTP	F
1	567	567	0	1
2	625	625	0	1
3	672	672	0	1
4	667	667	0	1
5	669	669	0	1
6	813	813	0	1
7	674	674	0	1
8	626	626	0	1
9	627	627	0	1
10	1100	1100+	0	1
11	1022	1022	0	1
12	1107.65	1100+	0.765	1
13	1107.56	1100	0.756	1
14	893	893	0	1
15	1107.56	1100+	0.756	1
16	1100	1100	0	1
17	800	800	0	1
18	1112.6	1100+	1.26	1
19	1100.4	1100	0.04	1
20	464	464	0	1
21	1107.56	1100+	0.756	1
22	1106.83	1100+	0.683	1
23	541	541	0	1
24	1107.56	1100+	0.756	1
25	1108.26	1100+	0.826	1
26	874	874	0	1
27	1112.6	1100+	1.26	1

Table D.34: Summary of test problem TP14(6).

Exp	PM	NE	DTP	F
1	1444	1444	0	1
2	8189	3800+	38.9	0
3	1514	1514	0	1
4	1099	1099	0	1
5	3706	3706	0	1
6	7898.5	3800+	9.85	0
7	1341.5	569	77.25	1
8	1348.5	576	77.25	1
9	1646.5	874	77.25	1
10	3800	3800+	0	1
11	1749	1749	0	1
12	3800	3800+	0	1
13	3800	3800+	0	1
14	1943	1943	0	1
15	3800	3800+	0	1
16	3800	3800+	0	1
17	3800	3800+	0	1
18	3800	3800+	0	1
19	3806.2	3800+	0.62	1
20	1528	1528	0	1
21	1514	1514	0	1
22	1099	1099	0	1
23	3806.2	3800+	0.62	1
24	3806.7	3800+	0.67	1
25	816	816	0	1
26	3806.1	3800+	0.61	1
27	3800.49	3800+	0.049	1

Table D.35: Summary of test problem TP15(2).

Exp	PM	NE	DTP	F
1	453	453	0	1
2	314	314	0	1
3	503.54	500+	0.354	1
4	4617.8	193	42.48	0
5	4622.8	198	42.48	0
6	4622.8	198	42.48	0
7	4617.8	193	42.48	0
8	4622.8	198	42.48	0
9	4622.8	198	42.48	0
10	49	49	0	1
11	48	48	0	1
12	50	50	0	1
13	48	48	0	1
14	48	48	0	1
15	19	19	0	1
16	48	48	0	1
17	48	48	0	1
18	19	19	0	1
19	23	23	0	1
20	23	23	0	1
21	23	23	0	1
22	20	20	0	1
23	20	20	0	1
24	19	19	0	1
25	20	20	0	1
26	20	20	0	1
27	19	19	0	1

Table D.36: Summary of test problem TP15(4).

Exp	PM	NE	DTP	F
1	1164.3	843	32.13	1
2	7274.1	2800+	47.41	0
3	790.1	260	53.01	1
4	7390	2800+	59	0
5	6805.54	2800+	0.554	0
6	7333.8	2800+	53.38	0
7	7441.2	2800+	64.12	0
8	7085.8	2800+	28.58	0
9	7448.9	2800+	64.89	0
10	796	796	0	1
11	668.2	626	4.22	1
12	935	935	0	1
13	2343	2343	0	1
14	2802.7	2747	5.57	1
15	1326.7	1271	5.57	1
16	1280.7	1174	10.67	1
17	1083.2	1041	4.22	1
18	706	706	0	1
19	1150	1150	0	1
20	770	770	0	1
21	1151.2	1109	4.22	1
22	2815.8	2800	1.58	1
23	779.2	737	4.22	1
24	644.7	589	5.57	1
25	1361	1361	0	1
26	839.2	797	4.22	1
27	1303	1303	0	1

Table D.37: Summary of test problem TP15(6).

Exp	PM	NE	DTP	F
1	2061	1491	57	1
2	23295.6	19000+	29.56	0
3	3310.5	2671	63.95	1
4	2428.8	1831	59.78	1
5	2004.3	1253	75.13	1
6	1056.8	449	60.78	1
7	19629.1	19000+	62.91	1
8	2309	1591	71.8	1
9	3570.5	2960	61.05	1
10	3307	2749	55.8	1
11	2512	1959	55.3	1
12	5728.6	4939	78.96	1
13	19025.4	19000+	2.54	1
14	3580.2	3116	46.42	1
15	4033.2	3470	56.32	1
16	6170.3	5867	30.33	1
17	4454.2	3708	74.62	1
18	3251.6	2485	76.66	1
19	13275.4	12834	44.14	1
20	1774.4	1436	33.84	1
21	2522.8	2215	30.78	1
22	18740.2	18219	52.12	1
23	2696	2696	0	1
24	2463.3	1732	73.13	1
25	4602	4602	0	1
26	4328.8	4034	29.48	1
27	2919	2548	37.1	1

Table D.38: Summary of test problem TP16(2).

Exp	PM	NE	DTP	F
1	206	206	0	1
2	273	273	0	1
3	216	216	0	1
4	260	260	0	1
5	317	317	0	1
6	340	340	0	1
7	4649.8	198	45.18	0
8	4649.8	198	45.18	0
9	4649.8	198	45.18	0
10	189.9	172	1.79	1
11	195.9	178	1.79	1
12	64.4	22	4.24	1
13	185.9	168	1.79	1
14	193.9	176	1.79	1
15	195.9	178	1.79	1
16	186.9	169	1.79	1
17	193.9	176	1.79	1
18	193.9	176	1.79	1
19	165.9	99	6.69	1
20	117.9	100	1.79	1
21	172.9	106	6.69	1
22	180	180	0	1
23	180	180	0	1
24	145	145	0	1
25	256	256	0	1
26	113.9	96	1.79	1
27	240	240	0	1

Table D.39: Summary of test problem TP16(4).

Exp	PM	NE	DTP	F
1	2850	2850	0	1
2	22594.9	18000+	59.49	0
3	790.1	260	53.01	1
4	22569.9	18000+	56.99	0
5	1604	1582	2.2	1
6	2422	2422	0	1
7	4492.7	4420	7.27	1
8	22197.79	18000+	19.779	0
9	22616.66	18000+	61.666	0
10	766.6	430	33.66	1
11	646	624	2.2	1
12	927	905	2.2	1
13	2254	2254	0	1
14	1604	1582	2.2	1
15	2422	2422	0	1
16	4492.7	4420	7.27	1
17	17989.7	17917	7.27	1
18	543.7	471	7.27	1
19	861	861	0	1
20	2158	2158	0	1
21	1297	1275	2.2	1
22	898	898	0	1
23	746.7	674	7.27	1
24	1383	1383	0	1
25	1163	1163	0	1
26	1400	1400	0	1
27	842	842	0	1

Table D.40: Summary of test problem TP16(6).

Exp	PM	NE	DTP	F
1	2061	1491	57	1
2	10764.2	6500+	26.42	0
3	4172.9	3579	59.39	1
4	2321.1	1721	60.01	1
5	1845.9	1362	48.39	1
6	2969.6	2804	16.56	1
7	2194.6	1574	62.06	1
8	2309	1591	71.8	1
9	4086.7	3381	70.57	1
10	3274.3	2648	62.63	1
11	6844.3	6299	54.53	1
12	4494.1	3767	72.71	1
13	3897.1	3383	51.41	1
14	5834.5	5798	3.65	1
15	2969.6	2804	16.56	1
16	4157.6	3995	16.26	1
17	3118.3	2371	74.73	1
18	4086.7	3381	70.57	1
19	3191.8	3157	3.48	1
20	2283.3	2233	5.03	1
21	4510.4	4393	11.74	1
22	4914.3	4908	0.63	1
23	4529	4529	0	1
24	2645.3	2547	9.83	1
25	1846.2	1175	67.12	1
26	4756.5	4746	1.05	1
27	2834	2541	29.3	1

Table D.41: Summary of test problem TP17(4).

Exp	PM	NE	DTP	F
1	1252	1252	0	1
2	23540	19000+	54	0
3	23081.2	19000+	8.12	0
4	23013.6	19000+	1.36	0
5	23137.9	19000+	13.79	0
6	23466.7	19000+	46.67	0
7	23360	19000+	36	0
8	23398.1	19000+	39.81	0
9	23038.5	19000+	3.85	0
10	4860.9	4823	3.79	1
11	717	717	0	1
12	280.3	265	1.53	1
13	18650.9	18095	55.59	1
14	241	241	0	1
15	500	500	0	1
16	19020	19000+	2	1
17	377	377	0	1
18	259	259	0	1
19	517	517	0	1
20	535.9	498	3.79	1
21	537	537	0	1
22	367	367	0	1
23	527	527	0	1
24	547	547	0	1
25	429.9	392	3.79	1
26	418.9	381	3.79	1
27	349	349	0	1

Table D.42: Summary of test problem TP18(4).

Exp	PM	NE	DTP	F
1	5576.6	1100+	47.66	0
2	5247.4	1100+	14.74	0
3	5158	1100+	5.8	0
4	679	679	0	1
5	5456	1100+	35.6	0
6	5502.1	1100+	40.21	0
7	594.7	575	1.97	1
8	679.7	660	1.97	1
9	492.7	473	1.97	1
10	504.7	485	1.97	1
11	1098.8	1077	2.18	1
12	334.6	292	4.26	1
13	632.7	613	1.97	1
14	750.7	731	1.97	1
15	783	783	0	1
16	397.7	371	2.67	1
17	442	442	0	1
18	553.7	534	1.97	1
19	571.7	552	1.97	1
20	575.7	549	2.67	1
21	450.7	431	1.97	1
22	678.7	659	1.97	1
23	422	422	0	1
24	605.7	586	1.97	1
25	411.7	385	2.67	1
26	427.7	408	1.97	1
27	995.7	969	2.67	1

Table D.43: Summary of test problem TP19(4).

Exp	PM	NE	DTP	F
1	4420	400+	2	0
2	4401.29	400+	0.129	0
3	4950.3	400+	55.03	0
4	4850.36	400+	45.036	0
5	4408.895	400+	0.8895	0
6	4771.57	400+	37.157	0
7	4400.652	400+	0.0652	0
8	4842.1	400+	44.21	0
9	4821.3	400+	42.13	0
10	279	279	0	1
11	621	621	0	1
12	172	172	0	1
13	334	334	0	1
14	277	277	0	1
15	166	166	0	1
16	171	171	0	1
17	165	165	0	1
18	336	336	0	1
19	309	309	0	1
20	289	289	0	1
21	261	261	0	1
22	525	525	0	1
23	284	284	0	1
24	267	267	0	1
25	300	300	0	1
26	317	317	0	1
27	358	358	0	1

Table D.44: Summary of test problem TP20(4).

Exp	PM	NE	DTP	F
1	2748	2748	0	1
2	8164.4	3700+	46.44	0
3	7733.3	3700+	3.33	0
4	3095	3095	0	1
5	3124	3124	0	1
6	8091.2	3700+	39.12	0
7	7708.3	3700+	0.83	0
8	7709.8	3700+	0.98	0
9	2969	2969	0	1
10	2013	2013	0	1
11	350.9	343	0.79	1
12	293.4	284	0.94	1
13	347.9	340	0.79	1
14	260.46	251	0.946	1
15	280	267	1.3	1
16	2025	2025	0	1
17	3068	3068	0	1
18	2969	2969	0	1
19	1799	1799	0	1
20	307.9	300	0.79	1
21	2800	2800	0	1
22	310.9	303	0.79	1
23	3632	3632	0	1
24	388.9	381	0.79	1
25	1924	1924	0	1
26	1944	1944	0	1
27	378.9	371	0.79	1

Appendix E

Copyright permissions

E.1 Permission to reprint definitions, figure and numerical examples from Brea and Cheng (2003a)

The copyright permission for including the definitions of Section 2, figure entitled "Figure 1. Flow chart of the LCNM algorithm" and the numerical examples of Section 6 from Brea and Cheng (2003a) was obtained by email, which is textually shown as follows:

Date: Thu, 4 Mar 2004 12:00:52 -0000

From: "Al-Dabass, David" <david.al-dabass@ntu.ac.uk>

To: Ebert Brea <E.Brea@maths.soton.ac.uk>

Subject: Permission for including...

Dear Mr Brea,

In my capacity as Editor of the Proceedings of UKSIM 2003 6th National Conference of the United Kingdom Simulation Society, held in Cambridge, UK, in April 2003, I here by grant:

Permission to include in your PhD thesis: the definitions of Section 2, figure entitled "Figure 1. Flow chart of the LCNM algorithm" and the numerical examples of Section 6.

From the paper by Brea, E and R. C. H. Cheng which appeared in the said proceedings as: Brea, E. and R. C. H. Cheng (2003), Constrained simulation

optimization, Proc. of UKSIM 2003 Sixth National Conference of the United Kingdom

Simulation Society, Cambridge, UK, April 2003, David Al-Dabass, ed., pp 99-105.

Yours Sincerely,

Professor David Al-Dabass

Chairman: UK Simulation Society,

Professor of Intelligent Systems

The Nottingham Trent University

Nottingham

NG1 4BU.

References

Anderson, E. J. and M. C. Ferris (2001). A direct search algorithms for optimization with noisy function evaluations. *SIAM Journal on Optimization 11*(3), 837–857.

Arnold, S. F. (1990). *Mathematical statistics* (First ed.). Englewood Cliffs, USA: Prentice Hall, Inc.

Audet, C. and J. E. Dennis Jr (2003). Analysis of generalized pattern searches. *SIAM Journal on Optimization 13*(3), 889–903.

Barton, R. R. and J. S. Ivey, Jr. (1996). Nelder-Mead simplex modifications simulation optimization. *Management Science 42*(7), 954–973.

Bazaraa, M. S. and C. M. Shetty (1979). *Nonlinear programming: Theory and algorithms* (1st ed.). New York, USA: John Wiley and Sons.

Boender, C. G. E. and H. E. Romeijn (1995). Stochastic methods. In R. Horst and P. M. Pardalos (Eds.), *Handbook of global optimization*, Volume 2 of *Nonconvex optimization and its applications*, pp. 829–869. Dordrecht ; Boston: Kluwer Academic Publishers.

Box, G. E. P. (1952). Multi-factor designs of first order. *Biometrika 39*(1/2), 49–57.

Box, G. E. P. (1957). Evolutionary operation: A method for increasing industrial productivity. *Applied Statistics 6*(2), 81–101. Royal Statistical Society.

Box, G. E. P. and K. B. Wilson (1951). On the experimental attainment of optimum conditions. *Journal of the Royal Statistical Society. Series B (Methodological) 13*(1), 1–45.

Brea, E. (2002). Identification of optimum operation in simulation models. Transfer thesis, University of Southampton, Faculty of Mathematical Studies, Department of Mathematics, Graduate School in Mathematical Studies.

Brea, E. and R. C. H. Cheng (2003a). Constrained simulation optimization. In D. Al-Dabass (Ed.), *UKSIM 2003 Sixth National Conference of the United Kingdom Simulation Society*, Cambridge, UK, pp. 99–105. United Kingdom Simulation Society.

Brea, E. and R. C. H. Cheng (2003b). Optimisation of four queue network via Nested Partition method. *International Journal of Simulation: Systems, Science and Technology 4*(1/2), 23–28.

Brooks, S. H. and M. R. Mickey (1961). Optimum estimation of gradient direction in steepest ascent experiments. *Biometrics 17*(1), 48–56.

Brown, M. B. and A. B. Forsythe (1974). Robust test for test equality of variances. *Journal of the American Statistical Association 69*(346), 364–367.

Chen, C.-H., J. Lin, E. Yücesan, and S. E. Chick (2000). Simulation budget allocation for further enhancing the efficiency of ordinal optimization. *Discrete Event Dynamic Systems: Theory and Applications 10*(3), 251–270.

Cheng, R. C. H. (2003a, 6 October 2003). Canonical form of the symmetrically linearly constrained linear programming problem.

Cheng, R. C. H. (2003b, May 2003). Slow convergence rate of the Linear Constrained Nelder-Mead method.

Corana, A., M. Marchesi, C. Martini, and S. Ridella (1987). Minimizing multimodal functions of continuous variables with the "Simulated Annealin" algorithm. *ACM Transaction on Mathematical Software 13*(3), 262–280.

Fu, M. C. (1994). Optimization via simulation: A review. In P. L. Hammer (Ed.), *Simulation and Modeling*, Volume 53 of *Annals of Operations Research*, pp. 199–247. Amsterdam, The Netherlands: J. C. Baltzer AG, Science Publishers.

Gill, P. E., W. Murray, and M. H. Wright (1991). *Numerical linear algebra and optimization* (1st ed.), Volume 1. Redwood City, California, USA: Addison-Wesley Publishing Conpany.

Hedlund, P. and A. Gustavsson (1998). Design and evaluation of an improved simplex method. *Analytica Chimica Acta 371*(1), 9–21.

Hock, W. and K. Schittkowski (1981). *Test examples for nonlinear programming codes.* Lecture notes in economics and mathematical systems ; 187. Berlin ; New York: Springer-Verlag.

Hollander, M. and D. A. Wolfe (1999). *Nonparametric statistical method* (Second ed.). New York: John Wiley and Sons, Inc.

Hooke, R. and T. A. Jeeves (1961). "Direct Search" solution of numerical and statistical problems. *Journal of the Association for Computing Machinery 8*(2), 212–229.

Hsu, J. C. (1996). *Multiple comparisons: Theory and methods* (First ed.). London: Chapman and Hall.

Humphrey, D. G. and J. R. Wilson (2000). A revised simplex procedure for stochastic simulation response surface optimization. *INFORMS Journal on Computing 12*(4), 272–283.

Kelley, C. T. (1999). Detection and remediation of stagnation in the Nelder-Mead algorithm using a sufficient decrease condition. *SIAM Journal on Optimization 10*(1), 43–55.

Khuri, A. I. and J. A. Cornell (1996). *Response surface: Designs and analyses* (Second ed.), Volume 152 of *Statistics: Textbooks and Monographs*. New York: Marcel Dekker, Inc.

Kolda, T. G., R. M. Lewis, and V. Torczon (2003). Optimization by Direct Search: New perspectives on some classical and modern methods. *SIAM Review 45*(3), 385–482.

Lagarias, J. C., J. A. Reeds, M. H. Wright, and P. E. Wright (1998). Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM Journal on Optimization 9*(1), 112–147.

Law, A. M. and W. D. Kelton (2000). *Simulation modeling and analysis* (3rd ed.). McGraw-Hill series in industrial engineering and management science. Boston: McGraw-Hill.

Lewis, R. M. and V. Torczon (1999). Pattern search algorithms for bound constrained minimization. *SIAM Journal on Optimization 9*(4), 1082–1099.

Lewis, R. M. and V. Torczon (2000). Pattern search methods for linearly constrained optimization. *SIAM Journal on Optimization 10*(3), 917–941.

Lewis, R. M., V. Torczon, and M. W. Trosset (2000). Direct search methods: then and now. *Journal of Computational and Applied Mathematics 124*(1-2), 191–207.

Lucidi, S. and M. Sciandrone (2002). On the global convergence of derivative-free methods for unconstrained optimization. *SIAM Journal on Optimization 13*(1), 97–116.

Lucidi, S., M. Sciandrone, and P. Tseng (2002). Objective-derivative-free methods for constrained optimization. *Mathematical Programming 92*(1), 37–59.

Mangasarian, O. L. (1969). *Nonlinear programming.* Systems Science. New York,: McGraw-Hill.

Mann, H. B. and D. R. Whitney (1974). On a test of whether one of two random variables is stochastically larger that the other. *The Annals of Mathematical Statistics 18*(1), 50–60.

Markushevich, A. I. (1965). *Theory of functions of a complex variable* (1st ed.), Volume 2. Englewood Cliffs, New Jersey, USA: Prentice-Hall International, INC.

McKinnon, K. I. M. (1998). Convergence of the Nelder-Mead simplex method to a nonstationary point. *SIAM Journal on Optimization 9*(1), 148–158.

Minitab (1994). *Minitab reference manual: Release 10 for Windows.* State College, PA, USA: Minitab Inc.

Minitab (2000). *Independent Two-Sample t-test.* http://www.minitabtraining.com/Downloads/materialsexample.pdf: Minitab Inc.

Moré, J. J., B. S. Garbow, and K. E. Hillstrom (1981). Testing unconstrained optimization software. *ACM Transaction on Mathematical Software 7*(1), 17–41.

Myers, R. H. and D. C. Montgomery (2002). *Response surface methodology: Process and product optimization using designed experiments.* Wiley Series in Probability and Statistics. New York: John Wiley and Sons, Inc.

Neddermeijer, H. G., N. Piersma, G. J. van Oortmarssen, J. D. F. Habbema, and R. Dekker (1999). Comparison of response surface methodology and the Nelder and Mead simplex method for optimization in microsimulation models. Technical Report Econometric Institute Report EI-9924/A, Erasmus University Rotterdam. Last access on 4 April 2004 at http://www.few.eur.nl/few/research/pubs/ei/1999/reports.htm.

Nelder, J. A. and R. Mead (1965). A simplex method for function minimization. *Computer Journal 7*, 308–313.

Nelder, J. A. and R. Mead (1966). A simplex method for function minimization-Errata. *Computer Journal 8*, 27.

Noble, B. and J. W. Daniel (1988). *Applied linear algebra* (3rd ed.). Englewood Cliffs, New Jersey, USA: Prentice-Hall International, INC.

Nocedal, J. and S. J. Wright (1999). *Numerical optimization.* Springer series in operations research. New York: Springer.

Olver, F. W. J. (1974). *Asymptotics and special functions* (1st ed.). New York, USA: Academic Press, INC.

Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal 7*(2), 155–162.

Price, C. J., I. D. Coope, and D. Byatt (2002). A convergent variant of the Nelder-Mead algorithm. *Journal of Optimization: Theory and Applications 113*(1), 5–19.

Rabinowitz, F. M. (1995). Algorithm 744: A stochastic algorithm for global optimization with constraints. *ACM Transaction on Mathematical Software 21*(2), 194–213.

Rosenbrock, H. H. (1960). An automatic method for finding the greatest or least value of a function. *The Computer Journal 3*(3), 175–184.

Routh, M. W., P. A. Swartz, and M. B. Denton (1977). Performance of the super modified simplex. *Analytica Chemistry 49*(9), 1422–1428.

Rubinstein, R. Y. (1992). *Monte Carlo optimization, simulation, and sensitivity of queuing networks* (Reprint ed.). Malabar, Fla.: Krieger Pub. Co. Reuven Y. Rubinstein. 24 cm. Original published: New York : Wiley, 1986.

Ryan, B. F., B. L. Joiner, and T. A. Ryan Jr (1985). *Minitab handbook* (Second ed.). Boston, USA: PWS-Kent Publishing Company.

Schittkowski, K. (1987). *More test examples for nonlinear programming codes.* Lecture notes in economics and mathematical systems ; 282. Berlin ; New York: Springer-Verlag.

Shi, L. and C.-H. Chen (2000). A new algorithm for stochastic discrete resource allocation optimization. *Discrete Event Dynamic Systems: Theory and Applications 10*(3), 271–294.

Shi, L. and S. Ólafsson (2000a). Nested Partitions method for global optimization. *Operations Research 48*(3), 390–407.

Shi, L. and S. Ólafsson (2000b). Nested Partitions method for stochastic optimization. *Methodology and Computing in Applied Probability 2*(3), 271–291.

Spendley, W., G. R. Hext, and F. R. Himsworth (1962). Sequential application of simplex designs in optimisation and evolutionary operation. *Technometrics 4*(4), 441–461.

Sphar, C. (1999). *Learn Microsoft Visual C++ 6.0 Now.* Redmond, WA: Microsoft Press. Microsoft Corporation, Microsoft Visual C++ 6.0 deluxe learning edition.

Subrahmanyam, M. B. (1989). An extension of the simplex method to constrained nonlinear optimization. *Journal of Optimization Theory and Applications 62*(2), 311–319.

Torczon, V. (1997). On the convergence of pattern search algorithms. *SIAM Journal on Optimization 7*(1), 1–25.

Trosset, M. W. (1997). I know it when I see it: toward a definition of direct search methods. *SIAG/OPT Views-and-News: A Forum for the SIAM Activity Group on Optimization* (9), 7–10. Fall 1997.

Walters, F. H., L. R. Parker Jr, S. L. Morgan, and S. N. Deming (1991). *Sequential simplex optimization : a technique for improving quality and productivity in research, development, and manufacturing.* Boca Raton, Fla.: CRC Press.