Solving Max-Cut Optimization Problem

Basim Rabaa Jumaa  
College of Education for Pure Sciences  
University of Babylon  
basim4345@gmail.com (07712276044)

Dr Ahmed Sabah Al-Jilawi  
College of Education for Pure Sciences  
University of Babylon  
aljelawy2000@yahoo.com (07719409850)

Abstract: The goal of this paper is to find a better method that converges faster of Max-Cut problem. One strategy is to the comparison between Bundle Method and the Augmented Lagrangian method. We have also developed the theoretical convergence properties of these methods.

Keywords: Max-Cut, Augmented Lagrangian Method, Bundle Methods and Constrained Optimization problems.

1. INTRODUCTION

Optimization is a primary mathematical method aimed at finding the value of variables that provide the minimum value for a mathematical function. Optimization algorithms are a basic and efficient technique in mathematical programming, to arrive at a solution, generally with the help of a computer. Optimization algorithms start with a first estimate of the value of the variables and by an iterative technique generates a sequence of get better estimates, or iterates, until an optimal solution is reached.

A great algorithm should be accurate, fast, efficient and robust. A good approximation of an optimal solution should be generated.

We here present a short overview of multiplier methods [1]. The beginning in the field of multiplier methods begins with Joh (1943). Kuhn and Tucker (1951) are eminent scientists who have conducted extensive research in the field of multiplier methods. Its results on the necessary conditions and adequate conditions are important in this field. Arrow and Hurwicz (1956) introduced the Lagrangian function [2]. King (1966) [3] developed the augmented Lagrangian algorithms. Hestenes and Powell [4] showed that the algorithm is locally convergent if the second-order sufficient conditions are satisfied. Miele et al. (1971) [5] and Rockafellar (1970) [6] introduced an augmented Lagrangian method for inequality constrained convex programming.

This method has been studied by Rockafellar in several papers [7]. The augmented Lagrangian method has got a powerful theoretical tool for convex programming. Arrow, Gould, and Howe (1971) [8] studied Rockafellar’s augmented Lagrangian method and Pierre (1971) [9] introduced a special augmented Lagrangian method with local convergence properties. This method was also studied by Lowe (1974) [10], Bertsekas (1982) [11], Humes (2000), R. A. Polyak (2001), R. A. Castillo (2003), J. M. Martinez (2006), S. Leyffer (2007) and H. Z. Luo et al. (2011).
Recently, many researchers have been interested in Lagrange’s enhanced methods, such as Leyffer (2016) [12], Kanzow et al. (2018) [13] and Lourenço (2018) [14]. The benefit of the augmented Lagrangian method is that it is robust, and we do not need a feasible beginning point. The augmented Lagrangian method has been used to solve optimization problems with both equality and inequality constraints [11].

Also, the Bundle method was independently created by Claude Lemarechal [15] and Philip Wolfe [16] in (1975). Since then a large number of variants of bundle methods have been developed, such as proximal bundle (1990) [17], trust region bundle (2001) [18]. Bundle methods are at the moment the most efficient and promising methods for smooth optimization and they have been successfully used in many practical applications, for example, in engineering, economics, mechanics and optimal control. (2002) [19]. The convergence of the minimization algorithm was studied and compare them with different versions of the bundle methods using the results of numerical experiments (2013) [20]. Bundle methods have been extensively studied to solve convex and nonconvex optimization problems (2015) [21]. The a simple version of the bundle method with linear programming is suggested. (2019) [22].

2. MAX-CUT Problem

The maximum cut (MAX-CUT) problem is an fascinating area of combinatorial optimization and has several applications in different fields, for instance, physics, computer science, and mathematics. This problem is NP-hard [23]. The abbreviation NP denoting for non-deterministic polynomial time, which means NP-hard is a difficult problem that can not be solved accurately. Several papers have studied the MAX-CUT problem. This line of research was started by Goemans-Williamson (1995) [24] with their approximation for the MAX-CUT problem based on semidefinite programming relaxation. Poljak showed that linear programming techniques cannot accomplish a better approximation solution [25], which is why semidefinite programming has attracted great importance and research activity. (for more details see [26]).

Suppose $G = (V, E)$ a non-directed graph is with the vertex set $V$ and edge set $E$, and suppose $w_{ij} = w_{ji}$ be edge weight $ij \in E$ and $w_{ij} = 0$ if $ij \notin E$. The adjacency matrix of $G$ is given by $A = [a_{ij}]$ such that $a_{ij} = w_{ij}$.

We can write the Max-Cut problem model as

\[
\begin{align*}
(MC) \quad \text{maximize} \quad & \sum_{i<j} w_{ij} \left( \frac{1-x_i x_j}{2} \right) \\
\text{subject to} \quad & x \in \{-1,1\}^n,
\end{align*}
\]

the objective function $\sum_{i,j \in E} w_{ij} \left( \frac{1-x_i x_j}{2} \right) = \text{ the number of cut edges.}$ For example, if

\[
x = \begin{pmatrix} 1 \\ 1 \\ 1 \\ -1 \end{pmatrix} \in \{-1,1\}^4.
\]

Then

\[
\begin{align*}
x_1 x_2 &= (-1)(1) = -1 \quad \rightarrow \quad \text{edge}(1,2) \text{ is cut} \\
x_2 x_3 &= (1)(1) = 1 \quad \rightarrow \quad \text{edge}(2,3) \text{ is not cut} \\
x_1 x_3 &= (-1)(1) = 1 \quad \rightarrow \quad \text{edge}(1,5) \text{ is not cut} \\
x_4 x_5 &= (1)(-1) = -1 \quad \rightarrow \quad \text{edge}(4,5) \text{ is cut}
\end{align*}
\]
3. The bundle and Augmented Lagrangian Methods

In the optimization problem, we wish to minimize or maximize some function subject to some constraint. The general problem of optimization given by [27, 28]:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{such that} & \quad x \in X.
\end{align*}
\]

The function \( f \) is defined from a convex set \( X \subseteq \mathbb{R}^n \) into \( \mathbb{R} \). A point \( x^* \in X \) is a local solution of problem (1) if there exists a neighborhood \( B(x^*, k) \) where \( f(x^*) \leq f(x) \) for every \( x \in B(x^*, k) \cap X = \{x \in X \mid \|x - x^*\| \leq k\} \).

3.1 Optimality Conditions for Unconstrained Optimization

In this section, we consider the problem of unconstrained optimization. If \( X = \mathbb{R}^n \), i.e., minimize \( f \) sans constraints [27, 28], it can be expressed by:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad x \in \mathbb{R}^n.
\end{align*}
\]

\( f \) is continuously differentiable, then a necessary condition for \( x^* \in \mathbb{R}^n \) is a solution of problem (2)

\[\nabla f(x^*) = 0.\]

If \( f \) is twice continuously differentiable, then a necessary condition for \( x^* \in \mathbb{R}^n \) is a solution of problem (2)

\[\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) \succeq 0.\]

The sufficient conditions for \( x^* \in \mathbb{R}^n \) is a local solution of problem (3)

\[\nabla f(x^*) = 0, \quad \nabla^2 f(x^*) > 0.\]

**Theorem 3.1 (First-Order Necessary Condition (FONC)) [28]**

Suppose \( f : \mathbb{R}^n \to \mathbb{R} \) be differentiable. If \( x^* \) is a local minimizer of \( f \), then \( \nabla f(x^*) = 0 \).

**Proof:** Define \( h: \mathbb{R} \to \mathbb{R} \) as \( h(\rho) = f(x^* + \rho w) \) for some \( w \in \mathbb{R}^n \), then

\[h'(\rho) = w^T \nabla f(x^* + \rho w).\]

If \( \rho = 0 \), then \( h'(0) = w^T \nabla f(x^*) \). By definition

\[h'(\rho) = \lim_{\rho \to 0} \frac{f(x^* + \rho w) - f(x^*)}{\rho}.\]

Since \( x^* \) is a local minimizer, there exists \( \epsilon > 0 \), where \( f(x^* + \rho w) \geq f(x^*) \) for every \( 0 < \rho < \epsilon \), thus we get \( w^T \nabla f(x^*) \geq 0 \). Since \( w \) is arbitrary, we can substitute \( w \) by \(-w\), and thus \( -w^T \nabla f(x^*) \geq 0 \). So, \( w^T \nabla f(x^*) = 0 \), for every \( w \in \mathbb{R}^n \). Thus, \( \nabla f(x^*) = 0 \).
Theorem 3.2 (Second-Order Necessary Condition (SONC)) [28]
Suppose \( f : \mathbb{R}^n \to \mathbb{R} \) be twice differentiable. If \( x^* \) is a local minimizer of \( f \), then \( \nabla f(x^*) = 0 \) and \( \nabla^2 f(x^*) \) is positive semidefinite.

Proof: Suppose \( w \in \mathbb{R}^n \). We want to prove that \( w^T \nabla^2 f(x^*) w \geq 0 \).

By using Taylor expansion of \( f \) at \( x^* \), we get
\[
f(x^* + \rho w) = f(x^*) + f'(x^*)\rho w + \frac{1}{2} \rho^2 \nabla^2 f(x^*) w + o(\rho^2).
\]
Since \( \nabla f(x^*) = 0 \) by \( \nabla \text{ONC} \) theorem, we have
\[
f(x^* + \rho w) = f(x^*) + \frac{1}{2} \rho^2 \nabla^2 f(x^*) w + o(\rho^2).
\]
Divide the sides on \( \rho^2 \), we have
\[
\frac{f(x^* + \rho w) - f(x^*)}{\rho^2} = \frac{1}{2} w^T \nabla^2 f(x^*) w + \frac{o(\rho^2)}{\rho^2}.
\]
We take the limit to both sides, and use the fact that \( x^* \) is a local minimizer, we get
\[
0 \leq \lim_{\rho \to 0} \frac{f(x^* + \rho w) - f(x^*)}{\rho^2} = \frac{1}{4} \lim_{\rho \to 0} \frac{1}{\rho^2} w^T \nabla^2 f(x^*) w + \frac{o(\rho^2)}{\rho^2}.
\]
Since
\[
\lim_{\rho \to 0} \frac{\rho_o(\rho^2)}{\rho^2} = 0,
\]
we conclude that \( w^T \nabla^2 f(x^*) w \geq 0 \). So, \( \nabla^2 f(x^*) \) is positive semidefinite.

Theorem 3.3 (Second-Order Sufficient Condition (SOSC)) [28]
Suppose \( f : \mathbb{R}^n \to \mathbb{R} \) be twice continuously differentiable. If \( \nabla f(x^*) = 0 \) and \( \nabla^2 f(x^*) \) is positive definite, then \( x^* \) is a strict local minimizer.

3.2. The Augmented Lagrangian Method
This method started to be used in the 1970s. Initially, it was called the multipliers method. Now, this method is called the augmented Lagrangian method. The goal of this method is to solve constrained optimization problems. This is done by substitute a constrained problem with a series of unconstrained problems [4]. The augmented Lagrangian method is analogous to the bundle method since in both of them a bundle term is added to the objective. The difference in the augmented Lagrangian method is the Lagrange multiplier term is added to it [27].

The augmented Lagrangian method was introduced by Hestenes [4]. To introduce the augmented Lagrangian method, we change the constraint \( h_i(x) = 0 \) to the constraint \( h_i(x) + \beta \alpha = 0 \), therefore, we get the problem
\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad h_i(x) + \beta \alpha = 0, \quad i = 1, \ldots, E, \quad \alpha \in \mathbb{R}. \\
\end{align*}
\]
We apply the bundle method to the problem (3), we get the augmented Lagrangian function as follows. We start with the bundle problem for the problem (3)
\[
\arg\min_x f(x) + \frac{1}{2\beta} (h(x) + \beta \alpha)^T (h(x) + \beta \alpha),
\]
this expands to
\[
\arg\min_x f(x) + \frac{1}{2\beta} (h(x)^T h(x) + 2\beta \alpha^T h(x) + \beta^2 \alpha^T \alpha),
\]
after simplification we get
\[
\arg\min_x f(x) + \alpha^T h(x) + \frac{1}{2\beta} \| h(x) \|^2.
\]
Thus, the augmented Lagrangian function is
\[
F(x, \alpha, \beta) = f(x) + \alpha^T h(x) + \frac{1}{2\beta} \| h(x) \|^2.
\]
We apply the equality augmented Lagrangian and the bundle methods to the linear programming (LP) problem. The initial results will provide an idea to work in semidefinite programming. The bundle method and augmented Lagrangian methods can be used for equality.

The generally LP problem is given as

\[ \begin{align*}
\text{minimize} & \quad \langle c, x \rangle \\
\text{subjectto} & \quad \langle a_i, x \rangle = b_i, \quad i = 1, \ldots, m \\
& \quad x \geq 0, \quad x \in \mathbb{R}^n.
\end{align*} \]

where \( a_i \in \mathbb{R}^n \), for \( i = 1, \ldots, m \), the numbers \( b_i \in \mathbb{R} \), for \( i = 1, \ldots, m \), and the vector \( c \in \mathbb{R}^n \), \( x_i \geq 0 \), for \( i = 1, \ldots, n \). The Lagrangian is given by

\[ \mathcal{L}(x, y) = \langle c, x \rangle + \langle y, b_i - \langle a_i, x \rangle \rangle \]

The general form for the augmented Lagrangian is

\[ L_a(x, y) = f(x) + \langle y, b_i - \langle a_i, x \rangle \rangle + \frac{1}{2a} \| b_i - \langle a_i, x \rangle \|^2. \]

Consider the primal \((P)\) and the dual \((D)\) standard linear programming problems:

\[ \begin{align*}
(P) \quad & \text{maximize} \quad \langle c, x \rangle \\
& \text{subjectto} \quad Ax = b \\
& \quad x \geq 0
\end{align*} \]

and

\[ \begin{align*}
(D) \quad & \text{minimize} \quad \langle b, y \rangle \\
& \text{subjectto} \quad A^T y \geq c
\end{align*} \]

Example 3.1 Consider the following simple linear programming problem:

\[ \begin{align*}
(P) \quad & \text{maximize} \quad x_1 + 2x_2 - 4x_3 \\
& \text{subjectto} \quad x_1 + x_2 + x_3 = 0 \\
& \quad x \geq 0
\end{align*} \] (4)

The dual of problem (4) is given by

\[ \begin{align*}
(D) \quad & \text{minimize} \quad 9y_1 \\
& \text{subjectto} \quad y_1 \geq 1 \\
& \quad y_1 \geq 2 \\
& \quad y_1 \geq 4
\end{align*} \] (5)

The optimal solution of problem (4) is \( x_1 = x_2 = 0, \ x_3 = 9 \) and the optimal value is \( x_1 + 2x_2 + 4x_3 = 36 \); an optimal solution of problem (5) is \( y_1 = 4 \) and the optimal value is \( 9y_1 = 36 \).

4. Algorithms and Numerical Computation

In this section, we discuss the numerical results of the algorithms by using Julia Language (JuliaBox). The numerical results were generated using the augmented Lagrangian method, which was validated with the bundle method. This test was done on a specific graph that was imported from the Biq Mac library [29] in Figure 2. This graph consists of 100 nodes connected with 2475 edges. The figure shows that the exact MAX-CUT solution equals 1430.
Figure 2: Biq Mac library (100 nodes and 2475 edges)
4.1. Augmented Lagrangian Methods

The multiplier method is to update the Lagrange Multiplier estimate [29] and sometimes the bundle parameter \( \beta \) in all iteration. The method of the multiplier is summarized in the Algorithm [1].

Algorithm [1] : The Augmented Lagrangian Methods
1. Choose \( x^0 \) and \( \beta^0 > 0 \), choose \( \alpha^0 \).
2. Find \( x^{k+1} \) such that
\[
F(x^{k+1}) = \min_x F(x, \alpha^k, \beta^k)
\]
3. Update \( \beta^k \) and \( \alpha^k \).
4. Set \( k = k + 1 \) and repeat.

4.2. Bundle Methods [30]

We define another method that can be considered as a stabilization of the plane’s cutting method.

We start by adding an additional point called the center, \( y^k \), to the bundle of information. We will continue to use the same linear model for our function \( f \), but it is no longer a solution \( \text{LP} \) on each iteration. Instead, we will compute the next iterate of the Algorithm [2].

Algorithm [2] : Bundle Method
1. Let \( \delta > 0, m \in (0.1), y^0, x^0 = y^0 \), and \( k = 0 \). Compute \( f(y^0) \).
2. Compute the next iterate
\[
x^{k+1} = \arg\min_{y \in \mathbb{R}^n} f_k(x) + \frac{\alpha}{2} \| x - y^k \|^2
\]
3. Define \( \delta_k = f(y^k) - [f_k(x^{k+1}) + \frac{\alpha}{2} \| x^{k+1} - y^k \|^2] \geq 0 \).
4. If \( \delta_k < 0 \) Stop.
5. Compute \( f(x^{k+1}) \).
6. Update the model
\[
\hat{f}_{k+1}(x) = \max\{f_k(x), f(x^{k+1}) - \langle s^{k+1}, x - x^{k+1} \rangle\}.
\]
7. Set \( k = k + 1 \) and go to Step 2.

4.3. Numerical Resultes

In this section, we will review our results and we are assessing the performance of the development algorithm proposed. The figures in this section illustrate the number of function calls of the approaches being used to solve the max-cut problems. Different sizes of cases were tested, results were extracted and shown in this section.

In Figure [3] and Figure [4], it is obvious that the Augmented Lagrangian Method provides a more rapid convergence. The bundle methods converge after 4s, while the augmented Lagrangian methods required only 3s for the convergence.
Figure 3: Augmented Lagrangian

Figure 4: The Bundle
CPU time may change between runs due to the use of other software on the same computer. Accordingly, it was more accurate to plot the number of function calls rather than the CPU time. Which cannot be affected by any other program that is run at the same time by using our program, it is evident that the augmented Lagrangian method performed faster and required fewer function calls, while the bundle method required more function calls.

4.4. L-BFGS and BFGS methods [31]

In the section, we present methods solve of minimizing the problem optimization [L-BFGS, BFGS and CQ methods]

- Limited-memory BFGS (L-BFGS) is an optimization algorithm in the quasi-Newton method family that is using a limited amount of computer memory to approximate the Broyden – Fletcher – Goldfarb – Shanno algorithm (BFGS).
- It is a common algorithm for parameter estimation in machine learning. The target problem for the algorithm is to minimize over unconstrained values for the real vector.
- The algorithm L-BFGS solves the problem of minimizing an objective, given its gradient, by iteratively measure approximations of the Hessian inverse matrix.
- The conjugate gradient (CQ) method is an algorithm for the numerical solution of specific linear equation systems, namely those whose matrix is symmetric positive-definite.

Table [1] and Table [2] reports the CPU time and number of function calls for several graphs in the Biq Mac library that have 100 nodes and 2475 edges [29].

| Problem  | Aug. (LBFGS) m=1 time | Aug. (LBFGS) m=2 time | Aug. (LBFGS) m=10 time | Aug. (BFGS) time | Aug. (CQ) time |
|----------|-----------------------|-----------------------|------------------------|-----------------|---------------|
|          | fcalls                | fcalls                | fcalls                 | fcalls          | fcalls        |
| g05_100.0| 1.22                  | 112                   | 1.31                   | 120             | 1.38          | 118           | 1.88          | 161           | 0.75          | 80            |
| g05_100.1| 1.19                  | 118                   | 1.22                   | 108             | 1.25          | 102           | 1.39          | 119           | 0.73          | 82            |
| g05_100.2| 1.24                  | 115                   | 1.31                   | 125             | 1.34          | 118           | 1.60          | 158           | 0.65          | 81            |
| g05_100.3| 1.37                  | 130                   | 1.27                   | 122             | 1.70          | 139           | 2.15          | 181           | 0.87          | 100           |
| g05_100.4| 1.80                  | 163                   | 1.68                   | 150             | 1.63          | 152           | 1.72          | 158           | 0.88          | 100           |
| g05_100.5| 1.44                  | 120                   | 1.50                   | 130             | 1.47          | 122           | 1.71          | 155           | 0.80          | 90            |
| g05_100.6| 1.30                  | 110                   | 1.30                   | 109             | 1.49          | 117           | 1.35          | 116           | 0.74          | 86            |
| g05_100.7| 1.25                  | 106                   | 1.20                   | 108             | 1.26          | 108           | 1.20          | 100           | 0.77          | 79            |
| g05_100.8| 1.43                  | 119                   | 1.45                   | 119             | 1.50          | 126           | 1.46          | 126           | 0.88          | 90            |
| g05_100.9| 1.46                  | 102                   | 1.30                   | 96              | 1.36          | 100           | 2.01          | 91            | 1.12          | 72            |

Table 1: CPU time and function calls number of iterations for augmented Lagrangian method
In figure [5] we plotted the bounds against the CPU time to compare the performance of the augmented Lagrangian methods and the bundle methods. This test was performed on a specific graph imported from the Biq Mac library [29]. It is evident that the augmented Lagrangian method performed faster and required 28 function calls to converge, while the bundle method required more than 37 function calls.

Table 2: CPU time and function calls number of iterations for bundle method

| Problem  | Bundle. (LBFGS) m=1 | Bundle. (LBFGS) m=2 | Bundle. (LBFGS) m=10 | Bundle. (BFGS) | Bundle. (CQ) |
|----------|----------------------|---------------------|----------------------|----------------|--------------|
| g05_100.0 | 2.65 240             | 2.50 233             | 2.46 234             | 3.21 302      | 1.44 161     |
| g05_100.1 | 2.00 204             | 2.01 202             | 2.40 203             | 3.31 255      | 1.05 133     |
| g05_100.2 | 2.30 250             | 2.52 260             | 2.82 261             | 3.18 355      | 1.58 187     |
| g05_100.3 | 2.85 295             | 2.65 270             | 2.79 270             | 5.44 444      | 1.65 218     |
| g05_100.4 | 3.15 338             | 3.00 329             | 3.40 322             | 5.27 515      | 2.00 234     |
| g05_100.5 | 2.50 266             | 2.68 282             | 2.58 267             | 4.02 416      | 1.70 192     |
| g05_100.6 | 3.00 234             | 2.88 240             | 2.37 242             | 3.22 340      | 1.47 160     |
| g05_100.7 | 2.26 238             | 2.10 212             | 2.21 222             | 2.72 281      | 1.18 150     |
| g05_100.8 | 2.61 267             | 2.26 234             | 2.55 239             | 3.81 355      | 1.81 186     |
| g05_100.9 | 3.00 229             | 2.82 218             | 2.75 216             | 3.77 315      | 1.82 149     |

Figure 5: Bounds vs CPU time for augmented Lagrangian and bundle methods.
5. The Theoretical Convergence Properties of the Bundle and Augmented Lagrangian Methods

In this section, we discuss the current major convergence theorems. We start with the convergence of the bundle and augmented Lagrangian methods. Recall that

\[ F(x, \beta) = f(x) + \frac{1}{\gamma} \| h(x) \|^2, \]

and, the augmented Lagrangian function is

\[ \mathcal{F}(x, \alpha, \beta) = f(x) + \alpha^T h(x) + \frac{1}{2 \beta} \| h(x) \|^2. \]

Suppose us define \( D(x) \) to be \( E \times n \) Jacobian of \( h(x) \), such that

\[ h(x) = [h_1(x), ..., h_E(x)]^T. \]

Hence

\[ D(x)^T = [\nabla h_1(x), ... \nabla h_E(x)]. \]

**Theorem 5.1 (Convergence of augmented Lagrangian method) [32]**

Suppose \( f \) and \( h \) be twice continuously differentiable functions. Suppose \( y^k = a^k + \frac{\gamma(x^k)}{\beta^k}, \)

and

\[ \| \nabla_y \mathcal{F}(x^k, \alpha^k, \beta^k) \| \leq \epsilon^k, \]

where \( \epsilon^k \to 0 \) as \( k \to \infty \). If \( x^k \) converges to \( x^* \), where \( \nabla h_i(x^*), i = 1, ..., E \), are linearly independent, then \( y^k \to y^* \) with \( y^* \) satisfying \( \nabla f(x^*) = D(x^*)^T y^* \). If additionally, either \( \beta^k \to 0 \) with bounded \( \alpha^k \) or \( \alpha^k \to y^* \) with bounded \( \beta^k \), then \( x^* \) satisfies the (FONC) and \( y^* \) is the vector of Lagrange multipliers.

**Proof:** Since the augmented Lagrangian function is

\[ \mathcal{L}(x, \beta, \alpha) = f(x) + \beta^T h(x) + \frac{1}{2 \alpha} \| h(x) \|^2, \]

we have that

\[ \nabla_x \mathcal{L}(x, \beta, \alpha) = \nabla f(x) + D(x)^T \beta + \frac{1}{\alpha} D(x)^T h(x). \]

It can be rewritten

\[ \nabla_x \mathcal{L}(x, \beta, \alpha) = \nabla f(x) + D(x)^T \left( \beta - \frac{h(x)}{\alpha} \right). \]

So, we have that

\[ \| \nabla_x \mathcal{L}(x^k, \beta^k, \alpha^k) \| = \| \nabla f(x^k) + h(x^k)^T y^k \| \leq \epsilon^k, \]

by assumption \( y^k \to y^* \), where \( y^* = -D(x^*)^T \nabla f(x^*) \), and that

\[ \nabla f(x^*) + D(x^*)^T y^* = 0. \]

Now, by definition of \( y^k \),

\[ \| h(x^k) \| = \alpha^k \| \beta^k - y^k \| \leq \alpha^k \| y^k - y^* \| + \alpha^k \| \beta^k - y^* \|. \]

By assumption, \( \alpha^k \to 0 \) with bounded \( \beta^k \) or \( \beta^k \to y^* \) with bounded \( \alpha^k \), so we have that \( h(x^k) \to 0 \) in either case. Since \( x^k \) converges to \( x^* \) and \( h \) is continuous, \( h(x^*) = 0 \). Thus \( (x^*, y^*) \) satisfies the (FONC).

**The Bundle Method [33]**

The aim is to provide the rate convergence of the bundle method for solving convex optimization problems of a form below

\[ \min_{x \in \mathbb{R}^n} F(x) \]  

and \( F: \mathbb{R}^n \to \mathbb{R} \) is a convex function.
The bundle method is linked to the basic idea of the proximal point method, who uses the Moreau-Yosida regularization for $F(.)$:

$$F_p(y) = \min_x \{ F(x) + \frac{\rho}{2} \| x - y \|^2 \}, \quad \rho > 0,$$

to build the proximal step for (6):

$$\text{prox}_F(y) = \arg\min_x \{ F(x) + \frac{\rho}{2} \| x - y \|^2 \}. \quad (6)$$

The proximal point method apply the iteration $x^{k+1} = \text{prox}_F(x^k)$, $k = 1, 2, ...$ and is converging to a minimum of $F(.)$, if a minimum exists [34]. The basic idea of the bundle method is to replace the problem (1) with a series of approximate problems of the following form:

$$\min_x F^k(x) + \frac{\rho}{2} \| x - y \|^2$$

Here $k = 1, 2, ...$ is the iteration number, $x^k$ is the best approximation to the solution, and $F^k(.)$ is a piecewise linear convex lower approximation of the function $F(.)$. Two versions of the method differ in the way it constructs this approximation.

**Theorem 5.2 (Convergence of bundle method):** [33]

Let $\text{Argmin} \ F \neq \emptyset$ and $\epsilon = 0$. Then a point $x^* \in \text{Argmin} \ F$ exists such that:

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

**Proof:** The proof of this result (in slightly different versions) it can be found in many references, such as [[34] Thm.4.9],[[35] Thm.XV.3.2.4], [[36] Thm.7.16]

6. Conclusions

The aim of the research has been achieved, and the following points have been clarified:

1- We test comparison between two methods the bundle method and Lagrangian augmentation method.
2- We prove the properties of theoretical convergence and we studied algorithms.
3- The graphs available in the Big Mac library were used to evaluate the method. These drawings included different features with a large number of edges and nodes
4- The results showed that the augmented Lagrangian method reached the goal in fewer the number of function calls of the bundle method and also was timed faster in CPU time.

References

[1] Judah Ben Rosen, Olvi L Mangasarian, and Klaus Ritter. *Nonlinear Programming: Proceedings of a Symposium Conducted by the Mathematics Research Center, the University of Wisconsin, Madison, May 4-6, 1970*. Number 25. Elsevier, 2014.
[2] Kenneth J Arrow and Leonid Hurwicz. Reduction of constrained maxima to saddle-point problems. In *Traces and Emergence of Nonlinear Programming*, pages 61–80. Springer, 2014.
[3] RP King. Necessary and sufficient conditions for inequality constrained extreme values. *Industrial & Engineering Chemistry Fundamentals*, 5(4):484–489, 1966.
[4] Magnus R Hestenes. Multiplier and gradient methods. *Journal of optimization theory and applications*, 4(5):303–320, 1969.
[5] Angelo Miele, EE Cragg, RR Iyer, and AV Levy. Use of the augmented penalty function in mathematical programming problems, part 1. *Journal of Optimization Theory and Applications*, 8(2):115–130, 1971.
[6] RT Rockafellar. New applications of duality in convex programming. In *Proceedings of the 4th Conference of Probability, Brasov, Romania*, pages 73–81, 1971.
[7] R Tyrell Rockafellar. The multiplier method of hestenes and powell applied to convex programming. *Journal of Optimization Theory and Applications*, 12(6):555–562, 1973.

[8] Kenneth Joseph Arrow, Floyd J Gould, and Stephen Mills Howe. A general saddle point result for constrained optimizations. Technical report, North Carolina State University. Dept. of Statistics, 1971.

[9] Pierre Ramond. Dual theory for free fermions. *Physical Review D*, 3(10):2415, 1971.

[10] Michael James Lowe. *Nonlinear programming: augmented Lagrangian techniques for constrained minimization*. PhD thesis, Montana State University-Bozeman, College of Engineering, 1974.

[11] Dimitri P Bertsekas. *Constrained optimization and Lagrange multiplier methods*. Academic press, 2014.

[12] Sven Leyffer and Charlie Vanaret. *Augmented lagrangian filter method*. 2016.

[13] Christian Kanzow, Daniel Steck, and Daniel Wachsmuth. An augmented lagrangian method for optimization problems in banach spaces. *SIAM Journal on Control and Optimization*, 56(1):272–291, 2018.

[14] Bruno F Lourenço, Ellen H Fukuda, and Masao Fukushima. Optimality conditions for nonlinear semidefinite programming via squared slack variables. *Mathematical Programming*, 168(1-2):177–200, 2018.

[15] Claude Lemarechal. An extension of davidon methods to non differentiable problems. In Nondifferentiable optimization, pages 95–109. Springer, 1975.

[16] Philip Wolfe. *A method of conjugate subgradients for minimizing nondifferentiable functions*. In Nondifferentiable optimization, pages 145–173. Springer, 1975.

[17] Krzysztof C Kiwiel. Proximity control in bundle methods for convex nondifferentiable minimization. *Mathematical programming*, 46(1-3):105–122, 1990.

[18] Stephan Dempe and Jonathan F Bard. *Bundle trust-region algorithm for bilinear bilevel programming*. Journal of Optimization Theory and Applications, 110(2):265–288, 2001.

[19] Marko Mäkelä. Survey of bundle methods for nonsmooth optimization. Optimization methods and software, 17(1):1–29, 2002.

[20] Kaisa Joki. *Nonsmooth optimization: Bundle methods*. 2013.

[21] Shuai Liu, Andrew Eberhard, and Yousong Luo. A version of bundle method with linear programming. arXiv preprint arXiv:1502.01787, 2015.

[22] Shuai Liu. A simple version of bundle method with linear programming. Computational Optimization and Applications, 72(2):391–412, 2019.

[23] Michael R Garey and David S Johnson. *Computers and intractability, vol. 29*, 2002.

[24] Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. Journal of the ACM (JACM), 42(6):1115–1145, 1995.

[25] Svatopluk Poljak and Zsolt Tuza. The expected relative error of the polyhedral approximation of the max-cut problem. Operations Research Letters, 16(4):191–198, 1994.

[26] Samuel Burer, Renato DC Monteiro, and Yin Zhang. Rank-two relaxation heuristics for max-cut and other binary quadratic programs. *SIAM Journal on Optimization*, 12(2):503–521, 2002.

[27] Dimitri P Bertsekas and Athena Scientific. *Convex optimization algorithms*. Athena Scientific Belmont, 2015.

[28] D Dimitri Bertsekas. *Nonlinear programming*. 1999. Athena, Scientific, Belmont CA.
[29] Angelika Wiegele. *Big mac library a collection of max-cut and quadratic 0-1 programming instances of medium size*. Preprint, 2007.

[30] Alexandre Belloni. *Lecture notes for iap 2005 course introduction to bundle methods*. Operation Research Center, MIT, Version of February, 11, 2005.

[31] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[32] N. Gould. A course on continuous optimization. [Accessed 3-12-2019].

[33] Yu Du and Andrzej Ruszczyński. *Rate of convergence of the bundle method*. Journal of Optimization Theory and Applications, 173(3):908–922, 2017.

[34] Claude Lemarechal. *Methods of descent for nondifferentiable optimization* (krzysztof c. kiwiel). SIAM Review, 30(1):146, 1988.

[35] JB Hiriart-Urruty and C Lemaréchal. *Convex analysis and minimization algorithms*. no. 305-306 in grund. der math. wiss, 1993.

[36] Andrzej Ruszczynski. *Nonlinear optimization*. Princeton university press, 2011.