Computational Experience with a Modified Augmented Lagrangian Merit Function in a Primal-Dual Interior-Point Method*

Draft
TR95-33

Z. Parada†  R. A. Tapia‡

September 28, 1995

Revisited July 1999

Abstract

In this paper we extend the classical Augmented Lagrangian function for the Equality Constrained Problem into the Nonlinear Programming problem based on ideas of Primal-Dual Interior-Point methods. We state basic properties on this Modified Augmented Lagrangian function in order to be a merit function for Newton's method applied to the perturbed Karush-Kuhn-Tucker conditions. A crucial point in this task is the choice of the penalty parameter, we suggest a rule to pick it up. Finally, we embed our Augmented Lagrangian Merit function in a Linesearch Newton Primal-Dual Interior-Point method and give numerical results in a subset of test problems

Keywords: Nonlinear Programming, Interior-Point Method, Augmented Lagrangian, Merit Function.

Abbreviated Title: Computational Experience.

1 Introduction

Recently, there has been much activity in interior-point method research for constrained optimization that investigates the advantages of using path-following strategies as well as different merit functions. Efforts toward this end include research in linear programming, convex programming, and nonlinear programming. It is well known that path-following strategies keep iterates away from the boundary. This is important since we know that iterates can "stick to the wall" and preclude the global convergence of interior point methods. However, preventing iterates from sticking to the wall may result in an expensive algorithm if we accurately satisfy the corresponding centrality conditions. A natural remedy for this problem is: choose a merit function that reflects the path-following strategy, and satisfies the centrality conditions in a nonrestrictive manner. Determining good path-following strategies and merit functions for constrained optimization from both a theoretical and computational standpoint is still an open research issue. However, a path-following

---

*Research sponsored by NSF Cooperative Agreement No. CCR-88-09615, ARO Grant 9DAAL03-90-G-0093, DOE Grant DEFG05-86-ER25017, and AFOSR Grant 89-0363.
†Graduate student, Department of Computational and Applied Mathematics, Rice University, Houston, Texas 77251-1892.
‡Professor, Department of Computational and Applied Mathematics, and Center of Research on Parallel Computation CRPC, Rice University, Houston, Texas, 77251-1892, e-mail: rat@rice.edu.
strategy should be an unnecessary luxury as El-Bakry, Tapia, Tsuchiya and Zhang [4] demonstrated recently in 1996 with their primal-dual interior-point method for nonlinear programming. Indeed, they proved global and local convergence theory for their method. In contrast, other authors have been quite successful in implementing and appropriate path-following strategies. Yamashita [12] in 1992 proposed an interior point method for constrained optimization by reducing his formulation to consist of only the primal variable. He was able to prove global convergence theory for his method. Also in 1992, Anstreicher and Vial [1] established a global primal-dual interior-point method for smooth convex programming. Their centrality conditions have the same flavor as the centrality conditions in linear programming. González-Lima [6], in 1994, found an effective manner to compute the analytic center solution for linear programming. González-Lima’s method is a modification of the Kojima-Mizuno-Yoshise’s algorithm [9] and the centrality condition is merely the $l_2$ norm of the duality gap. In 1995, Argáez and Tapia [2], defined a modified augmented Lagrangian as merit function of their primal-dual interior-point method for nonlinear programming. A novelty in their approach is the centrality condition strongly influenced by the augmentation. This centrality condition ignores information of the Lagrangian function. Impressive numerical results are reported for their method.

Following some of the ideas presented in the papers mentioned above, we define a modified augmented Lagrangian for nonlinear programming with the same flavor as the Argáez-Tapia modification. The augmentation is the $l_2$ norm of the equality constraints and perturbed complementary equation of the Karush-Kuhn-Tucker condition. Since our modified augmented Lagrangian satisfies the same basic properties as the Argáez-Tapia modification does, we decided to study the numerical behavior of our augmented Lagrangian as a merit function in an interior point method. This is the purpose of the present paper. The paper is organized as follows. In Section 2 we describe the nonlinear programming problem. In Section 3 we define the modified augmented Lagrangian. In Section 4 we describe a primal-dual interior-point method using the modified augmented Lagrangian as merit function. Basically this framework is the same as given by González-Lima [6] and Argáez-Tapia [2]. In Section 5 we detail the implementation of the method. In Section 6 we present our numerical experience of the method described in Section 4. Finally, in Section 7 we make concluding remarks.

2 The Nonlinear Programming

We will consider the standard problem

\[
\begin{align*}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & h(x) = 0 \\
\quad & x \geq 0,
\end{align*}
\]  

(2.1)

where $f : \mathcal{R}^n \rightarrow \mathcal{R}$, and $h : \mathcal{R}^n \rightarrow \mathcal{R}^m$, are twice continuously differentiable functions and $m \leq n$.

The Lagrangian function associated to problem (2.1) is given by

\[
l(x, y, z) = f(x) + h(x)^T y - x^T z,
\]

(2.2)

where $y \in \mathcal{R}^m$, and $z \in \mathcal{R}^n$, $z \geq 0$, are the Lagrange multipliers associated to the constraints $h(x) = 0$, and $x \geq 0$ respectively.

In order to simplify our writing throughout this paper, we introduce the following notation and function: the vector $w = (x, z, y)$ denotes all group of primal and dual variables in (2.2), and $v = (x, z)$ indicates the nonnegative primal and dual variables. Similar notation will be use in others vectors.

2
We also define the function $G : \mathcal{R}^{2n} \to \mathcal{R}^{m+n}$ given by

$$
G(v) = \begin{pmatrix}
    h(x) \\
    XZe
\end{pmatrix},
$$

(2.3)

where $X = \text{diag}(x)$, $Z = \text{diag}(z)$, and $e \in \mathcal{R}^n$ is the vector of all ones.

The Karush-Kuhn-Tucker (KKT) conditions of (2.1) are established as follows:

$$
F(w) = \begin{pmatrix}
    \nabla_x l(w) \\
    G(v)
\end{pmatrix} = 0, \quad v \geq 0.
$$

(2.4)

For $\mu > 0$, the perturbed KKT conditions associated to (2.4) are defined as:

$$
F_\mu(w) = \begin{pmatrix}
    \nabla_x l(w) \\
    G_\mu(v)
\end{pmatrix} = 0, \quad v > 0,
$$

(2.5)

where

$$
G_\mu(v) = G(v) - \mu \begin{pmatrix}
    0 \\
    e
\end{pmatrix}.
$$

(2.6)

Primal-Dual Interior-Point methods solving (2.1) approximates solutions of (2.5) by Newton’s method which corresponding iterates on $v$ remain positive. Once an approximated solution of (2.5) is obtained, the perturbation parameter $\mu$ is driving to zero. By doing so, it is expected to achieve the KKT conditions (2.4). Since Newton’s method is in general no globally convergent (see Dennis, and Schnabel[3]), there is need of using a merit function to globalize the above steps. The choice of merit function represents a significant scheme in a primal-dual interior-point method. The merit function may opt to approach the KKT conditions (2.4) or a path-following conditions based on the perturbed KKT conditions (2.5). The first option potentially leads to stick the boundaries given by the equations, $x_k s_k = 0$, $k = 1, \ldots, n$, in early iterations precluding Newton’s method. The second option prevents to approach these boundaries at a costly method when asking for too much accuracy in the path-following conditions.

Problem (2.1) is a nonlinear programming problem, hence a point $w$ satisfying the perturbed KKT conditions (2.5) may not exists for some $\mu > 0$. In fact Fiacco and McCormick[5], in 1966 demonstrated under mild assumptions at a local minimizer of (2.1) that (2.5) defines a local path. In order to stay away of the boundaries mentioned above, we consider the relaxed central conditions given by Argáez and Tapia[2] in 1995 that are described for a fixed $\mu > 0$ as the set:

$$
C_\mu = \{v \in \mathcal{R}^{2n} \mid G_\mu(v) = 0, \quad v > 0\}.
$$

(2.7)

These relaxed central conditions are merely copies for each positive perturbed parameter $\mu$ of the relative interior of the constraints in (2.1) given by the set $\mathcal{F}^0 = \{x \in \mathcal{R}^n \mid h(x) = 0, \quad x > 0\}$. According to the interior point philosophy, we assume that $\mathcal{F}^0$ is nonempty, then $C_\mu$ is well defined for any positive $\mu$.

Our effort will be to define a merit function so that the Newton step from (2.5) approaches the set $C_\mu$.

### 3 A Modified Augmented Lagrangian Function

In this section we describe the merit function to be used in the interior point method of section 4.
We define the Modified Augmented Lagrangian function associated to the perturbed parameter \( \mu \geq 0 \) as

\[
\phi_\mu(w; C) = l(w) + \frac{C}{2} \psi_\mu(v),
\]

(3.1)

where \( l(w) \) is the Lagrangian function given in (2.2), \( C \geq 0 \) is our penalty parameter, and the function

\[
\psi_\mu(v) = G_\mu(v)^T G_\mu(v), \quad \text{where} \quad v > 0,
\]

(3.2)

is the augmented term.

As it is common, we never differentiate (3.1) with respect to \( C \).

Observe that (3.1) is a generalization of the popular Augmented Lagrangian function for the equality constrained problem, see Hestenes [7]. Clearly by definition of (3.2), we have \( \psi_\mu(v) \geq 0 \), with equality if and only if \( v \in \mathcal{C}_\mu \).

We will demonstrate how to use our Modified Augmented Lagrangian function (3.1) as a merit function for Newton's method applied to the perturbed KKT conditions (2.5).

We begin by writing out the corresponding Newton linear system at the point \( w = (v, y)^T \), with \( v > 0 \) as follows:

\[
\begin{pmatrix}
\nabla_v (\nabla_x l(w)) & \nabla h(x) \\
G'_\mu(v) & 0
\end{pmatrix}
\begin{pmatrix}
\Delta v \\
\Delta y
\end{pmatrix}
= - \begin{pmatrix}
\nabla_v l(w) \\
G_\mu(v)
\end{pmatrix}.
\]

(3.3)

To clarify our discussion, we assume that the linear system (3.3) has unique solution on \( \Delta w = (\Delta v^T, \Delta y^T) \).

It is worth noticing from (3.3) that the reduced step \( \Delta v \) satisfies the unsquared linear system \((m + n \text{ equations}, 2n \text{ variables})\) given by

\[
G'_\mu(v) \Delta v = - G_\mu(v).
\]

(3.4)

Equation (3.4) is a right Newton's method applied to the nonlinear equation, \( G_\mu(v) = 0 \). This interpretation allows to establish the descent direction property on \( \Delta v \) and the augmented term (3.2).

Let's observe that

\[
\nabla \psi_\mu(v) = 2 \left[ G'_\mu(v) \right]^T G_\mu(v).
\]

Thus

\[
\nabla \psi_\mu(v)^T \Delta v = 2 G_\mu(v)^T G'_\mu(v) \Delta v,
\]

by using (3.4), we finally obtain

\[
\nabla \psi_\mu(v)^T \Delta v = -2 \psi_\mu(v) \leq 0,
\]

(3.5)

with equality if and only if \( v \in \mathcal{C}_\mu \).

Throughout this section, we assume that the inequality (3.5) will be strict at current point \( w = (v, y) \), i.e., \( v \notin \mathcal{C}_\mu \).
The relation (3.5) plays a central role to propose (3.1) as a merit function. We concentrate our effort to state descent direction property for the reduced Newton step, $\Delta v$, in the modified augmented Lagrangian function (3.1) rather than the full step $\Delta w$. For updating the variable $y$ in the method, we will use either the full Newton step $\Delta y$ or the damping step determined for $\Delta v$.

Let's begin computing the derivative of (3.1) in the direction of $\Delta v$. Then, we have

$$\nabla_v \phi_{\mu}(w; C)^T \Delta v = \nabla_v l(w)^T \Delta v + \frac{C}{2} \nabla \psi_{\mu}(v)^T \Delta v,$$

by (3.5) we obtain

$$\nabla_v \phi_{\mu}(w; C)^T \Delta v = \nabla_v l(w)^T \Delta v - C \psi_{\mu}(v).$$

Under our assumption of $v$, we are able to select a positive penalty parameter $C$ such that (3.6) is negative. However, choosing arbitrary large positive penalty parameter seems to be bad strategy as it is known in the Augmented Lagrangian function for equality constrained problem.

We are conservative and impose a condition in the penalty parameter. Since we deal with directional derivatives, we intent to select a penalty parameter in order to keep at least the derivative of $\psi_{\mu}(v)$ in the direction of $\Delta v$.

Our choice of the penalty parameter is based in the following simple observation. Since the Modified Augmented Lagrangian function (3.1) is composed by the Lagrangian function (2.2) and the augmented term (3.2), we will consider a penalty parameter that guarantee descent direction for the sum of these functions. It is clear that $\Delta v$ could not be a descent direction for (2.2) but it will for the negative of the Lagrangian function as long as the derivative is nonzero. More specifically, we consider the penalty parameter $C$ as the unique solution of the one dimensional linear equation given by

$$\nabla_v \phi_{\mu}(w; C)^T \Delta v = -|\nabla_v l(w)^T \Delta v| + \nabla \psi_{\mu}(v)^T \Delta v.$$  

We interprete the entire rate of descent direction (3.8) as the sum of rates of descent directions of each function that compose (3.1). The right hand side of (3.8) is negative under assumption, $v \not\in C_{\mu}$ . Equation (3.8) is independent of positive scaling of $\Delta v$. Invoking the basic steps of Section 1, we observe that the penalty parameter $C$ can be computed after either step S1 or step S2.

The solution of (3.8) is the positive number given by

$$C = 2 \left\{ \frac{|\nabla_v l(w)^T \Delta v|}{\psi_{\mu}(v)} + 1 \right\},$$

where $|r|_+$ indicates the real function whose value is either zero if $r$ is negative or $r$ itself otherwise.

We point out that if $\Delta v$ is already a descent direction of the Lagrangian function (2.2) then the penalization on the augmented term (3.2) in the Modified Augmented Lagrangian function (3.1) is one.

4 Primal-Dual Interior-Point Methods

In this section we present the primal-dual interior-point.
In the last section, we demonstrated that the reduced Newton step, $\Delta v$ is a descend direction for the Modified Augmented Lagrangian function (3.1). Furthermore, we established equation (3.8) to choose the penalty parameter. These components will cement our method.

The method for solving the KKT conditions (2.4) will consist of a path-following framework where the centrality condition are given by the set $C_\mu$. Proximity to $C_\mu$ will be measured by $\psi_\mu(v)$. As $\mu$ reduces values to zero, the method will generate iterates on $w = (v, y)$ so that the variable $v$ will follow the centrality conditions. To avoid a costly method we will find a point $w = (v, y)$ so that $v > 0$, and $\psi_\mu(v) \leq \gamma \mu$, where $\gamma$ is a constant in $(0, 1)$. Once a such point $w$ is obtained, the value of $\mu$ will be reduced to continue the process until the point $w$ satisfies our stopping criteria for the KKT conditions. The idea of shrinking neighborhoods for path-following methods as been used before in linear programming (see[6]) and nonlinear programming (see[12]). The Newton's step from the nonlinear equation $F_\mu(w) = 0$, will serve as tool to approach the centrality condition as we demonstrated above. Our Modified Augmented Lagrangian function (3.1) will be the merit function in the linesearch step. We have collected all the ingredients to establish the method. Recall that $F(w)$ is the residual function given in (2.4).

Algorithm 1 (Path-Following Primal-Dual Interior-Point Method)

Start: Let $w_0 = (v_0, z_0)$ be such that $v_0 = (x_0, z_0) > 0$. Fix $p, \beta, \gamma \in (0, 1)$.

Set $k = 0$ and $\mu_{-1} = 0$.

Step 1. Test for convergence using $\| F(w_k) \|_2$

Step 2. Choose $\sigma_k \in (0, 1)$ and set $\mu_k = \sigma_k \psi_{\mu_{k-1}}(v_k)$.

Step 3. Set $l = 0$, and $w^l = w_k$.

Step 4. (Inner Loop) If $\psi_{\mu_k}(v^l) \leq \gamma \mu_k$ go to Step 5.

4.1 Find $\Delta w^l$ as a solution of

$$F'_{\mu_k}(w^l)\Delta w^l = -F_{\mu_k}(w^l).$$

4.2. Compute $C^l$ as (3.9).

4.3. Choose $\tau^l \in (0, 1)$ and set $\bar{\alpha}^l = \min (1, \tau^l \bar{\alpha}^l)$ where

$$\bar{\alpha}^l = \min \left\{ \frac{-1}{\min((V^l)^{-1}\Delta v, -1)} \right\}.$$

4.4. Find the first natural number $s$ such that, $\alpha^l = p^s \bar{\alpha}^l$ satisfies

$$\phi_{\mu_k}(v^l + \alpha^l \Delta v^l, y^l, C^l) \leq \phi_{\mu_k}(w^l; C^l) + \beta \alpha^l \nabla \phi_{\mu_k}(w^l; C^l) ^T \Delta v^l.$$

4.5. Set $w^{l+1} = w^l + \alpha^l \Delta w^l$.

4.6. $l \leftarrow l + 1$, and go to Step 4.

Step 5. Set $w_{k+1} = w^l$.

Step 6. $k \leftarrow k + 1$, and go to Step 1.
Algorithm 1 generates two different classes of iterates. One class, called \( w^i \), is the path-following strategy defined by steps 4.1 to 4.6. Its goal is to approximate a solution for the centrality conditions. The second class is the outer iterates \( w_k \) in which the satisfaction of the KKT conditions (2.4) is tested in step 1.

The parametric choices are \( \sigma_k \) and \( \tau^i \). The parameter \( \sigma_k \) denotes how much centering we expect in the next outer iterate, and the steplength parameter \( \tau^i \) maintains positive iterates on \( v^{i+1} \).

For \( k \geq 1 \), a successful step 5 establishes that

\[
\psi_{\mu_{k-1}}(v_k) \leq \gamma \mu_{k-1}. \tag{4.1}
\]

Then, if step 1 fails, next step 2 ensures by (4.1) that

\[
\mu_k < \gamma \mu_{k-1}. \tag{4.2}
\]

Thus, our choice in step 2 yields convergence of \( \{\mu_k\} \) to zero. Furthermore, inequality (4.2) gives \( Q \)-linear rate of convergence.

In step 4.3, the parameter \( \rho^i \) is the smallest steplength for the components of \( v^i \) that reaches the non-negative boundary in the direction of \( \Delta v^i \). This steplength has become standard in most of interior-points methods.

Step 4.4 implements a backtracking scheme for the linesearch strategy.

For updating \( y^i \), we should open the choice of the full Newton step \( \Delta y^i \). Then, step 4.5 will be rewrote as:

**Step 4.5.** Set \( v^{i+1} = v^i + \rho^i \Delta v^i \), and \( y^{i+1} = y^i + \Delta y^i \).

As we are interested on numerical behavior of Algorithm 1, we will not concern in this paper about assumptions or properties ensuring successful termination of the inner loop.

## 5 Implementation

In this section we detail the steps on Algorithm 1 as well as the parametric choices.

We implement our method in Matlab 4.2 using a Sun workstation with 64 bits arithmetic.

The stopping criteria in step 1 is set as

\[
\frac{\|F(w_k)\|_2}{1 + \|w_k\|_2} \leq \epsilon_{exit} = 10^{-8}.
\]

The centering parameter in step 2 is the constant

\[
\sigma_k = 0.5.
\]

In the stopping criteria for the inner loop we set the constant

\[
\gamma = 0.8.
\]

The first and second order derivatives need are computed by forward finite difference.

We solve the Newton linear system in step 4.1 as follows:

First, we consider the reduced linear system
\[
\begin{pmatrix}
\nabla^2 l(u^i) + (X^i)^{-1}Z^i & \nabla h(x^i) \\
\n\nabla h(x^i)^T & 0
\end{pmatrix}
\begin{pmatrix}
\Delta x^i \\
\Delta y^i
\end{pmatrix}
= -\begin{pmatrix}
\nabla^2 l(u^i) + z^i - \mu_k(X^i)^{-1}e \\
h(x^i)
\end{pmatrix},
\]
and solved by Matlab subroutines. Then we take
\[
\Delta z^i = -(z^i - \mu_k(X^i)^{-1}e + (X^i)^{-1}Z^i\Delta x^i).
\]
In step 4.3, we set as constant the steplength parameter
\[r^i = .995.\]
For the linesearch strategy, we use the constants
\[p = 0.5, \text{ and } \beta = 10^{-4}.\]

6 Numerical Experience

In this section, we document our numerical experience with Algorithm 1. First we present our numerical results by means of tables showing behavior of Algorithm 1. Second we comment about performance of our parametric choices given in section 5.

6.1 Numerical Results

The test problems are from Hock and Schittkowski [8], and Schittkowski [11].

First, we compare the role of the centrality condition, \(\psi_\mu(v) = 0\), on our modified augmented Lagrangian merit function (3.1). We also implemented Algorithm 1 without the path-following strategy in step 4. It results in a damped and perturbed Newton's method applied to the KKT conditions (2.4). We called this version Algorithm 2. In Algorithm 2, the Newton's iteration is performed once for each \(\mu_k\). We left all parametric choices for Algorithm 2 as those we set in Algorithm 1. It is worth noticing that Algorithm 2 did not lead the sequence \(\{\mu_k\}\) to converge to zero as Algorithm 1 outlined. We summarize our numerical results in tables 1 and 2. Both tables are formed by six columns as follows: The first column contains the problem number. We labeled these problems with the same numbers they have in [8], and [11].

The second column is the dimension of the primal variable \(x\) (called \(n\)). The third and fourth columns are the number of equality constraints \((m)\) and inequality constraints \((p)\) respectively. The fifth and sixth columns are the number of linear system solved for Algorithm 1 and Algorithm 2 respectively. The starting points for the primal variable are the same as those in [8] and [11]. For the dual variables, we start with the vector of all ones. We solved 60 problems for Algorithm 1, and Algorithm 2 failed in 8 of these problems. For Algorithm 1, in 40 problems we found the solution reported in [8] and [11]. In some problems, we found better reported solutions In most of the test problems the number of linear system using Algorithm 1 or Algorithm 2 are similar. But the use of both algorithms produced different iterates. This behavior is shown in problems 81 and 104. These two problems are not solved by pure Newton method, i.e. the Algorithm 2 without backtracking step. For problem 81, and problem 104, we plot the following information: In horizontal axes are the number of the linear systems used for both algorithms; in vertical axes are the \(l_2\) norm of the KKT condition for each point generated by both algorithms. Solid line represented performance of Algorithm 1 and thin line did for Algorithm 2. See figure 1 and figure 2.

We observe that Algorithm 1 decreased the norm of the KKT condition faster that Algorithm 2 far away of the solution.

Second, we also investigated the behavior of the penalty parameter for Algorithm 1 and Algorithm 2. We summarize our results in table 3. The first column represents the first 30 problem. The second column means
the order of the last penalty parameter \( C \) for Algorithm 1. The third column did the same than second column, but using Algorithm 2. Starting column four until column six, we recicled information from the three previous columns. Fourth column are the number problems, the next two columns correspond to the order of the last penalty parameter using Algorithm 1 or Algorithm 2, respectively. Algorithm 2 gave in general smaller penalty parameters than Algorithm 1. This emphasized the role of the centrality condition which may force larger penalty parameters. We solved Problem 13 where the constrained qualifications does not hold. Problem 13 has been difficult to solve for interior-point codes (see El-Bakry et al [4], and Yamashita [12]).

6.2 Comments

In this section we comment about our numerical experience of Algorithm 1.

At outer iteration \( k \geq 1 \), the stopping criteria given in step 4 for inner iterate \( l \) is merely

\[
\psi_{\mu_k}(v^l) \leq \gamma \sigma_k \psi_{\mu_{k-1}}(v^l),
\]

then, small values of \( \sigma_k \), required too much accuracy for solving \( \psi_{\mu_k}(v) = 0 \), and Algorithm 1 became a costly method. Also values of \( \sigma_k \) close to 1 produced short steps in the satisfaction of the centrality conditions, hurting convergence of Algorithm 1. For \( \sigma_k \in [0.4, 0.6] \), our numerical results are much the same. Therefore we chose \( \sigma_k = 0.5 \). We moved parameter \( \gamma \) on interval \((0, 1)\). Values of \( \gamma \) close to zero enforced a costly method for Algorithm 1 as \( \sigma_k \) did. Also the term \( \gamma \sigma_k \) has to be large once \( \sigma_k \) is fixed. As \( \sigma_k = 0.5 \), the choice of \( \gamma \) must to be near to one. We recall that the task of \( \gamma \) is to guarantee a convergence sequence of perturbation parameter to zero. Experimentally, we chose the value of \( \gamma \) between

\[
0.5, 0.6, 0.7, 0.8, 0.9.
\]

We found the choice \( \gamma = 0.8 \) as the parameter that solved the maximum numbers of test problems regardless the number of linear systems to solve for each test problem.

The updating penalty parameter (3.9) suggested large values of \( C \), as \( \psi_{\mu}(v) \) is near zero. This behavior was enlisted in table 3. We can force the sequence of penalty parameter \( \{C^l\} \) to be monotone nondecrease by taking the largest value between \( C^{l-1} \) and the corresponding formula (3.9) at iteration \( l \). The choice of \( C^l \) ensures descent direction property for \( \Delta v^l \) on \( \psi_{\mu^l}(w : C^l) \) at \( w^l \). Numerical last value of \( C^l \) with this update was of the same order. We kept the update giving in (3.9).

We also considered different values of \( \tau^l \). If \( \tau^l = 0.9, 0.95, 0.99 \), Algorithm 1 took short steps to the boundary and convergence failed in some test problems.

7 Summary

We presented a Modified Augmented Lagrangian Function for the nonlinear programming problem (2.1) following ideas of interior-point methods. The augmentation term was merely the \( l_2 \) norm of the perturbed KKT conditions without Lagrangian information. We proved descent direction for the reduced Newton step into the Modified Augmented Lagrangian function by choosing an appropriate penalty parameter. Furthermore we proposed a primal-dual path-following interior-point method that captures our augmentation term. Finally, we tested our method in a subset of test problems obtaining promissory numerical results. Convergence theory of the method is a future work.

Acknowledgment. The first author would like to thank to the Department of Computational and Applied Mathematics at Rice University during his graduate studies. Also he thank the Center for Research on Parallel Computing at Rice University and the Instituto Tecnológico Autónomo de México in México City.
for support in the summer of 1999 to review this work. First author is currently Professor at: Departamento de Matemáticas, Instituto Tecnológico Autónomo de México, Rio Hondo 1, Col. Tizapán 01000, México D. F., México, e-mail: zeferino@gauss.rhon.itam.mx.
Figure 1: The norm of the KKT conditions for the two strategies on Problem 81

Figure 2: The norm of the KKT conditions for the two strategies on Problem 104
<table>
<thead>
<tr>
<th>Problem</th>
<th>n</th>
<th>m</th>
<th>p</th>
<th>Linear Systems</th>
<th>Alg 1</th>
<th>Alg 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>22</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>20</td>
<td>33</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>8</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>9</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>0</td>
<td>4</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>13</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>10</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>0</td>
<td>3</td>
<td>25</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>18</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>29</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>0</td>
<td>6</td>
<td>16</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>26</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>13</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>9</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>0</td>
<td>9</td>
<td>15</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>12</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>3</td>
<td>0</td>
<td>6</td>
<td>10</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>3</td>
<td>0</td>
<td>7</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>3</td>
<td>0</td>
<td>7</td>
<td>9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>3</td>
<td>1</td>
<td>4</td>
<td>12</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>3</td>
<td>0</td>
<td>8</td>
<td>14</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>27</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>3</td>
<td>0</td>
<td>7</td>
<td>15</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>3</td>
<td>0</td>
<td>8</td>
<td>17</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>4</td>
<td>0</td>
<td>8</td>
<td>23</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>4</td>
<td>1</td>
<td>8</td>
<td>12</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>4</td>
<td>0</td>
<td>10</td>
<td>14</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>4</td>
<td>0</td>
<td>10</td>
<td>11</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Hock and Schittkowski test problems. The symbol '-' means no convergence.
<table>
<thead>
<tr>
<th>Problem</th>
<th>n</th>
<th>m</th>
<th>p</th>
<th>Linear Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Alg 1</td>
</tr>
<tr>
<td>45</td>
<td>5</td>
<td>0</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>53</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>9</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>15</td>
</tr>
<tr>
<td>62</td>
<td>3</td>
<td>1</td>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>64</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>24</td>
</tr>
<tr>
<td>65</td>
<td>3</td>
<td>0</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>66</td>
<td>3</td>
<td>0</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>71</td>
<td>4</td>
<td>1</td>
<td>9</td>
<td>11</td>
</tr>
<tr>
<td>72</td>
<td>4</td>
<td>0</td>
<td>10</td>
<td>23</td>
</tr>
<tr>
<td>73</td>
<td>4</td>
<td>1</td>
<td>6</td>
<td>12</td>
</tr>
<tr>
<td>74</td>
<td>4</td>
<td>3</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>75</td>
<td>4</td>
<td>3</td>
<td>10</td>
<td>21</td>
</tr>
<tr>
<td>76</td>
<td>4</td>
<td>0</td>
<td>7</td>
<td>11</td>
</tr>
<tr>
<td>80</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>81</td>
<td>9</td>
<td>13</td>
<td>13</td>
<td>10</td>
</tr>
<tr>
<td>83</td>
<td>5</td>
<td>0</td>
<td>16</td>
<td>38</td>
</tr>
<tr>
<td>86</td>
<td>5</td>
<td>0</td>
<td>15</td>
<td>44</td>
</tr>
<tr>
<td>93</td>
<td>6</td>
<td>0</td>
<td>8</td>
<td>27</td>
</tr>
<tr>
<td>104</td>
<td>8</td>
<td>0</td>
<td>22</td>
<td>15</td>
</tr>
<tr>
<td>227</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>233</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>16</td>
</tr>
<tr>
<td>250</td>
<td>3</td>
<td>0</td>
<td>8</td>
<td>17</td>
</tr>
<tr>
<td>251</td>
<td>3</td>
<td>0</td>
<td>7</td>
<td>14</td>
</tr>
<tr>
<td>262</td>
<td>4</td>
<td>1</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>325</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td>339</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>341</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>9</td>
</tr>
<tr>
<td>342</td>
<td>3</td>
<td>0</td>
<td>4</td>
<td>17</td>
</tr>
<tr>
<td>353</td>
<td>4</td>
<td>1</td>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>354</td>
<td>4</td>
<td>0</td>
<td>5</td>
<td>14</td>
</tr>
</tbody>
</table>

Table 2: Hock and Schittkowski test problem (Continued). The symbol '−' means no convergence.
<table>
<thead>
<tr>
<th>Problem</th>
<th>Penalty Parameter</th>
<th>Problem</th>
<th>Penalty Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Alg 1</td>
<td>Alg 2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>2</td>
<td>45</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2</td>
<td>53</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>60</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>50</td>
<td>62</td>
</tr>
<tr>
<td>5</td>
<td>$10^3$</td>
<td>70</td>
<td>64</td>
</tr>
<tr>
<td>10</td>
<td>2.1</td>
<td>–</td>
<td>65</td>
</tr>
<tr>
<td>11</td>
<td>3.2</td>
<td>14.5</td>
<td>66</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>2</td>
<td>71</td>
</tr>
<tr>
<td>13</td>
<td>$10^8$</td>
<td>$10^8$</td>
<td>72</td>
</tr>
<tr>
<td>14</td>
<td>15.5</td>
<td>11</td>
<td>73</td>
</tr>
<tr>
<td>16</td>
<td>$10^4$</td>
<td>–</td>
<td>74</td>
</tr>
<tr>
<td>18</td>
<td>20</td>
<td>19</td>
<td>75</td>
</tr>
<tr>
<td>20</td>
<td>700</td>
<td>$10^4$</td>
<td>76</td>
</tr>
<tr>
<td>21</td>
<td>90</td>
<td>250</td>
<td>80</td>
</tr>
<tr>
<td>22</td>
<td>6.5</td>
<td>14</td>
<td>81</td>
</tr>
<tr>
<td>23</td>
<td>60</td>
<td>17</td>
<td>83</td>
</tr>
<tr>
<td>24</td>
<td>$10^4$</td>
<td>150</td>
<td>86</td>
</tr>
<tr>
<td>25</td>
<td>250</td>
<td>40</td>
<td>93</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>180</td>
<td>104</td>
</tr>
<tr>
<td>30</td>
<td>400</td>
<td>50</td>
<td>227</td>
</tr>
<tr>
<td>31</td>
<td>800</td>
<td>45</td>
<td>233</td>
</tr>
<tr>
<td>32</td>
<td>$10^3$</td>
<td>$10^3$</td>
<td>250</td>
</tr>
<tr>
<td>34</td>
<td>30</td>
<td>11</td>
<td>251</td>
</tr>
<tr>
<td>35</td>
<td>$10^3$</td>
<td>$10^3$</td>
<td>262</td>
</tr>
<tr>
<td>36</td>
<td>$10^4$</td>
<td>$10^4$</td>
<td>325</td>
</tr>
<tr>
<td>37</td>
<td>$10^3$</td>
<td>$10^4$</td>
<td>339</td>
</tr>
<tr>
<td>38</td>
<td>$10^5$</td>
<td>$10^7$</td>
<td>341</td>
</tr>
<tr>
<td>41</td>
<td>$10^3$</td>
<td>500</td>
<td>342</td>
</tr>
<tr>
<td>43</td>
<td>250</td>
<td>–</td>
<td>353</td>
</tr>
<tr>
<td>44</td>
<td>$10^3$</td>
<td>200</td>
<td>354</td>
</tr>
</tbody>
</table>

Table 3: The role of the centrality condition on the penalty parameter.
References


