Some NP-complete Problems in Quadratic and Nonlinear Programming

Katta G. Murty*

Dept. of Industrial and Operations Engineering
The University of Michigan
1205 Beal Avenue
Ann Arbor, MI 48109-2117, USA

Santosh N. Kabadi

Faculty of Administration
University of New Brunswick
Fredericton, NB, Canada E3B 5A6

Technical Report 85-23

June 1985

*Research Partially supported by NSF Grant No. ECS-8401081

ABSTRACT: The motivation for this study is to analyze the complexity of the problems of checking whether a given feasible solution is not a local minimum, and that of checking whether the objective function is not bounded below on the set of feasible solutions, in continuous variable, smooth, nonconvex nonlinear programming problems. We construct a very special instance called the simplest nonconvex NLP in this class (this is an indefinite quadratic programming problem with very simple objective and very simple constraints) with integer data and show that checking either of the above problems on this instance is an NP-complete problem. As a corollary we obtain the result that the problem of checking whether a given integer square matrix is not copositive, is NP-complete.

KEYWORDS: Convex or nonconvex nonlinear programming problems, local minimum, global minimum, unboundedness of the objective function, computational complexity, NP-completeness, indefinite quadratic programming, copositive matrices, positive semidefinite matrices.

ABBREVIATED TITLE: NP complete QPs.

INTRODUCTION

Consider the smooth nonlinear program (NLP)

$$\begin{aligned} &\text{minimize } \ \Theta(x) \\ &\text{subject to } \ g_i(x) \geqq 0, \ i = 1 \text{ to } m \end{aligned} \qquad (1)$$

where each of the functions is a real valued function defined on R^n, with high degrees of differentiability. Here $x = (x_1,\dots,x_n)^T \in R^n$. This NLP is said to be a convex NLP if $\Theta(x)$ is convex and $g_i(x)$ are concave for all i, nonconvex NLP, otherwise. Without any loss of generality, we assume that the NLP is always stated in minimization form.

For a convex NLP, under some constraint qualifications, necessary and sufficient optimality conditions are known. See [1, 3, 5]. Given a feasible solution satisfying the constraint qualification, it is possible to check very efficiently, using these optimality conditions, whether that point is a (global) optimum solution of the problem or not.

For nonconvex NLPs, under constraint qualifications, some necessary conditions for a local minimum are known, and there are some sufficient conditions for a point to be a local minimum. See [1, 3, 5]. But there are no simple conditions known, which are both necessary and sufficient for a given point to be a local minimum. The complexity of checking whether a given feasible solution is a local minimum in a nonconvex NLP is not usually addressed in the literature. When they discuss algorithms, many text books in NLP leave the reader with the impression that these algorithms converge to a global minimum in convex NLPs, and to a local minimum in nonconvex NLPs. The documentations for many professional software packages for NLP also create the same impression. This impression could be quite erroneous. In this paper, we study this problem by examining the computational complexity of determining whether a given feasible solution is not a local minimum, and that of determining whether the objective function is not bounded below on the set of feasible solutions, in smooth continuous variable nonconvex NLPs. For this purpose, we analyze an indefinite quadratic program (QP) with integer data, which may be considered as the simplest nonconvex NLP. It turns out that the questions of determining whether a given feasible solution is not a local minimum in this problem, and to check whether the objective function is not bounded below in this problem, can both be studied using the discrete techniques of computational complexity theory, and in fact these questions are NP-complete. This clearly shows that in general, it is a hard problem to check whether a given feasible solution in a nonconvex NLP is a local minimum or not, or to check whether the objective function is bounded below or not on its set of feasible solutions. This indicates the following: when a nonlinear programming algorithm is applied on a nonconvex NLP, unless it is proved that it converges to a point satisfying some known sufficient condition for a local minimum, claims that it leads to a local minimum are hard to verify in the worst case. Also, in continuous variable smooth nonconvex minimization, even the down-to-earth goal of guaranteeing that a local minimum will be obtained by an algorithm (as opposed to the lofty goal of finding the global minimum) may be hard to attain. We make some more comments on this issue at the end of the paper.

FINDING A GLOBAL MINIMUM IN A SMOOTH NONCONVEX NLP IS A HARD PROBLEM

The problem of computing a global minimum, or checking whether a given feasible solution is a global minimum, for a smooth nonconvex NLP, may be hard problems in general. We provide two examples.

EXAMPLE 1: FERMAT'S LAST CONJECTURE: Some of the hardest unsolved problems in mathematics can be posed as problems of finding a global minimum in a smooth nonconvex NLP. Consider the Fermat's last conjecture, unresolved since 1637 A.D. It states that there exists no positive integer solution (x, y, z) to the equation

$$x^n + y^n = z^n$$

when n is an integer ≥ 3 (here, x, y, $z \in R^1$). Eventhough this conjecture has been shown to be true for several individual values of n, in general, it remains open. Obviously, Fermat's last conjecture is true iff the global minimum objective value in the following NLP is > 0, where α is a positive penalty parameter, and Π is the length of the circumference of the circle of diameter 1 in R^2.

$$\text{minimize } (x^n + y^n - z^n)^2 + \alpha\Big((1 - \text{Cos}(2\Pi x))^2 + (1 - \text{Cos}(2\Pi y))^2$$
$$+ (1 - \text{Cos}(2\Pi z))^2 + (1 - \text{Cos}(2\Pi n))^2\Big)$$
$$\text{subject to } x, y, z \geq 1, n \geq 3.$$

EXAMPLE 2: SUBSET-SUM PROBLEM: To establish mathematically that the problem of computing a global minimum in a nonconvex NLP is a hard problem, we consider the subset-sum problem, a hard problem in discrete optimization, which is known to be NP-complete (see [2]): given positive integers d_0; $d_1, \dots, d_n$; is there a solution to

$$\sum_{j=1}^{n} d_j y_j = d_o \tag{2}$$
$$y_j = 0 \text{ or } 1 \text{ for all } j$$

Now consider the following quadratic programming problem

$$\text{minimize } \Big(\sum_{j=1}^{n} d_j y_j - d_o\Big)^2 + \sum_{j=1}^{n} y_j(1-y_j) \tag{3}$$
$$\text{subject to } 0 \leq y_j \leq 1, \ j = 1 \text{ to } n.$$

Because of the second term in the objective function, (3) is a nonconvex quadratic programming problem. Clearly (2) has a feasible solution iff the global minimum objective value in (3) is zero. Since the problem of checking whether (2) has a feasible solution is NP-complete, computing the global minimum in (3), a very special and simple case of a smooth nonconvex NLP, is an NP-hard problem. This shows that in general, the problem of computing a global minimum in a smooth nonconvex NLP may be a hard problem.

CAN WE AT LEAST COMPUTE A LOCAL MINIMUM FOR A SMOOTH NONCONVEX NLP, EFFICIENTLY?

Since the problem of computing a global minimum in a nonconvex NLP is a hard problem, we will now study the question of whether it is at least possible to compute a local minimum, or check whether a given feasible solution for such a problem is not a local minimum, by efficient algorithms. We review the known optimality conditions for a given feasible solution $\bar{x}$ to (1) to be a local minimum. Let $\mathbf{A} = \{i: \; g_i(\bar{x}) = 0\}$. Optimality conditions are derived under the assumption that some constraint qualifications (CQ, see [1, 3, 5]) are satisfied at $\bar{x}$, which we assume.

FIRST ORDER NECESSARY CONDITIONS FOR $\bar{x}$ TO BE A LOCAL MINIMUM FOR (1)

There must exist a $\bar{\mu}_{\mathbf{A}} = (\bar{\mu}_i: \; i \in \mathbf{A})$ such that

$$\nabla\theta(\bar{x}) - \sum_{i \in \mathbf{A}} \bar{\mu}_i \nabla g_i(\bar{x}) = 0 \qquad (4)$$

$$\bar{\mu}_i \geqq 0, \text{ for all } i \in \mathbf{A}$$

Here $\nabla\theta(\bar{x})$, $\nabla g_i(\bar{x})$ are the gradient vectors (row vectors) of these functions evaluated at $\bar{x}$. Given the feasible solution $\bar{x}$, it is possible to check whether (4) holds, efficiently, using Phase I of the simplex method for linear programming.

SECOND ORDER NECESSARY CONDITIONS FOR $\bar{x}$ TO BE A LOCAL MINIMUM FOR (1)

These conditions include (4). Given $\bar{\mu}_{\mathbf{A}}$ satisfying (4) together with $\bar{x}$, let $L(x, \bar{\mu}_{\mathbf{A}}) = \theta(x) - \sum_{i \in \mathbf{A}} \bar{\mu}_i g_i(x)$. In addition to (4) these conditions require

$$y^T H y \geqq 0, \text{ for all } y \in \{y: \; \nabla g_i(\bar{x}) y = 0 \text{ for each } i \in \mathbf{A}\} \qquad (5)$$

where H is the Hessian matrix of $L(x, \bar{\mu}_{\mathbf{A}})$ with respect to x at $x = \bar{x}$. Condition (5) requires the solution of a quadratic program involving only equality constraints, which can be solved efficiently. It is equivalent to checking the positive semidefiniteness of a matrix which can be carried out efficiently using Gaussian pivot steps (see [5]).

SUFFICIENT CONDITIONS FOR $\bar{x}$ TO BE A LOCAL MINIMUM FOR (1)

Given the feasible solution $\bar{x}$, and $\bar{\mu}_{\mathbf{A}}$ which together satisfy (4), the most general known sufficient optimality condition states that if

$$y^T H y > 0 \text{ for all } y \in T_1 \tag{6}$$

where $T_1 = \{y\colon\ y \neq 0 \text{ and } \nabla g_i(\bar{x})y = 0 \text{ for each } i \in \{i\colon\ i \in \mathbf{A} \text{ and } \bar{\mu}_i > 0\}, \text{ and } \nabla g_i(\bar{x})y \geqq 0 \text{ for each } i \in \{i\colon\ i \in \mathbf{A} \text{ and } \bar{\mu}_i = 0\}\}$. Unfortunately, when H is not positive semidefinite, the problem of checking whether (6) holds leads to a nonconvex QP, which, as we will see later, may be hard to solve.

Aside from the question of the difficulty of checking whether (6) holds, we can verify that the gap between conditions (5) and (6) is very wide, particularly when the set $\{i\colon\ i \in \mathbf{A} \text{ and } \bar{\mu}_i = 0\} \neq \emptyset$. In this case, condition (5) may hold, and even if we are able to check (6), if it is not satisfied, we are unable to determine whether $\bar{x}$ is a local minimum for (1) with present theory.

Now we will use a simple indefinite QP, related to the problem of checking whether the sufficient optimality condition (6) holds, to study the following questions.

i) Given a smooth nonconvex NLP and a feasible solution for it, can we check whether it is a local minimum or not efficiently?

ii) At least in the simple case when the constraints are linear, can we check efficiently whether the objective function is bounded below or not on the set of feasible solutions?

Let D be an integer square symmetric matrix of order n. D is said to be positive semidefinite (PSD) if $x^T D x \geqq 0$ for all $x \in R^n$. So checking whether D is not PSD involves the question "is there an $x \in R^n$ satisfying $x^T D x < 0$". It is well known that this question can be settled by performing at most n Gaussian pivot steps along the main diagonal of D, requiring a computational effort of at most $O(n^3)$. (For example, see [5 or 6]).

The matrix D is said to be copositive if $x^T Dx \geqq 0$ for all $x \geqq 0$. All PSD matrices are copositive, but a matrix which is not PSD may be copositive. Testing whether the given matrix D is not copositive involves the question "is there an $x \geqq 0$ satisfying $x^T Dx < 0$". If D is not PSD, no efficient algorithm for this question is known (some enumerative methods are available for this problem, see [5], but the computational effort required by these methods grows exponentially with n in the worst case). In fact we show later that this question is NP-complete. To study this question, we are naturally led to the NLP

$$\begin{aligned} &\text{minimize} \quad Q(x) = x^T Dx \\ &\text{subject to} \quad x \geqq 0 \end{aligned} \tag{7}$$

We will show that this problem is an NP-hard problem.

We assume that D is not PSD. So Q(x) is nonconvex and (7) is a nonconvex NLP. It can be considered the simplest nonconvex NLP. We consider the following decision problems.

Problem 1: Is $x = 0$ not a local minimum for (7)?

Problem 2: Is Q(x) not bounded below on the set of feasible solutions of (7)?

Clearly, the answer to problem 2 is in the affirmative iff the answer to problem 1 is. We will show that both these problems are NP-complete. To study problem 1, we can replace (7) by the NLP

$$\begin{aligned} &\text{minimize} \quad Q(x) = x^T Dx \\ &\text{subject to} \quad 0 \leqq x_j \leqq 1,\ j = 1 \text{ to } n \end{aligned} \tag{8}$$

LEMMA 1: The decision problem "is there an $\bar{x}$ feasible to (8) which satisfies $Q(\bar{x}) < 0$", is in the class NP.

PROOF: Given an $\bar{x}$ feasible to (8), to check whether $Q(\bar{x}) < 0$, can be done by computing $Q(\bar{x})$ which takes $O(n^2)$ time. Also, if the answer to the problem is in the affirmative, an optimum solution $\bar{x}$ of (8) satisfies $Q(\bar{x}) < 0$. There is a linear complementarity problem (LCP) corresponding to (8) and an optimum solution for (8) must correspond to a basic feasible solution (BFS) for this LCP. Since there are only a finite number of BFSs for an LCP, and they are all rational vectors, a nondeterministic algorithm can find one of them satisfying $Q(x) < 0$, if it exists, in polynomial time. Hence, this problem is in the class NP. ■

LEMMA 2: The optimum objective value in (8) is either 0 or $\leqq -2^{-L}$ where L is the size of D.

PROOF: Since the set of feasible solutions of (8) is a compact set and $Q(x)$ is continuous, (8) has an optimum solution.

By well known results, the necessary optimality conditions for (8) lead to the following linear complementarity problem (LCP), see [5 or 6].

$$\begin{pmatrix} u \\ \cdots \\ v \end{pmatrix} - \left(\begin{array}{c:c} D & I \\ \hdashline -I & 0 \end{array}\right) \begin{pmatrix} x \\ \cdots \\ y \end{pmatrix} = \begin{pmatrix} 0 \\ \cdots \\ e \end{pmatrix} \tag{9}$$

$$\begin{pmatrix} u \\ \cdots \\ v \end{pmatrix} \geqq 0, \quad \begin{pmatrix} x \\ \cdots \\ y \end{pmatrix} \geqq 0, \tag{10}$$

$$\begin{pmatrix} u \\ \cdots \\ v \end{pmatrix}^T \begin{pmatrix} x \\ \cdots \\ y \end{pmatrix} = 0 \tag{11}$$

where y is the column vector of Lagrange multipliers associated with the constraints "$x_j \leqq 1$ for all j", and u, v are the column vectors in R^n of dual and primal slack variables. For every optimum solution x of (8), there exist vectors u, v, y such that (u, v, x, y) solves (9), (10) and (11). Also, it can be verified that whenever (u, v, x, y) satisfies (9), (10) and (11), $x^TDx = -e^Ty$, a linear function, where e is the column vector of all 1's in R^n. So, there exists an optimum solution of (8) which is a basic feasible solution (BFS) of (9), (10). By the results under the ellipsoid algorithm (see, for example [4, 5]), in every BFS of (9), (10), each y_j is either 0 or $\geqq 2^{-L}$. If the optimum objective value in (8) is not zero, it must be < 0, and this together with the above facts implies that an optimum solution x of (8) corresponds to a BFS (u, v, x, y) of (9), (10) in which $-e^Ty < 0$. All these facts clearly imply that the optimum objective value in (8) is either 0 or $\leq -2^{-L}$.

We now make a list of several decision problems, some of which we have already seen, and some new ones which we need for establishing our results.

Problem 3: Is there an $x \geq 0$ satisfying $Q(x) < 0$?

Problem 4: For any positive integer a_0, is there an $x \in R^n$ satisfying $e^Tx = a_0$, $x \geq 0$ and $Q(x) < 0$?

Now consider a subset sum problem with data d_0; $d_1, \ldots, d_n$, which are all positive integers. Let γ be a positive integer $> 4(d_0(\sum_{j=1}^n d_j))^2 n^3$. Let ℓ be the size of this subset sum problem, that is, the total number of digits in all the data for the problem. Let ε be a positive rational number $< 2^{-n\ell^2}$. The subset sum problem is:

Problem 5: Subset sum problem: Is there a $y = (y_j) \in R^n$ satisfying $\sum_{j=1}^n d_j y_j = d_0$, $0 \leq y_j \leq 1$, $j = 1$ to n, and y integer vector.

We now define several functions involving nonnegative variables $y = (y_1, \ldots, y_n)^T$ and $s = (s_1, \ldots, s_n)^T$, related to the subset sum problem.

$$f_1(y, s) = (\sum_{j=1}^n d_j y_j - d_0)^2 + \gamma \left(\sum_{j=1}^n (y_j + s_j - 1)^2\right) + \sum_{j=1}^n y_j s_j$$

$$= (\sum_{j=1}^n d_j\ y_j)^2 + \sum_{j=1}^n y_j s_j + \gamma \sum_{j=1}^n\ (y_j + s_j)^2$$

$$-2d_0(\sum_{j=1}^n d_j y_j) - 2\gamma \sum_{j=1}^n (y_j + s_j) + n\gamma + d_0^2$$

$$f_2(y, s) = f_1(y, s) + 2d_0\left(\sum_{j=1}^n d_j y_j (1 - y_j)\right)$$

$$= (\sum_{j=1}^n d_j y_j)^2 + \gamma \sum_{j=1}^n (y_j + s_j)^2 + \sum_{j=1}^n y_j s_j$$

$$-2d_0(\sum_{j=1}^n d_j y_j^2) - 2\gamma \sum_{j=1}^n (y_j + s_j) + n\gamma + d_0^2$$

$$f_3(y, s) = (\sum_{j=1}^{n} d_j y_j)^2 + \gamma \sum_{j=1}^{n} (y_j + s_j)^2 + \sum_{j=1}^{n} y_j s_j$$

$$-2d_o \sum_{j=1}^{n} d_j y_j^2 + d_o^2 - n\gamma.$$

$$f_4(y, s) = (\sum_{j=1}^{n} d_j y_j)^2 + \gamma \sum_{j=1}^{n} (y_j + s_j)^2 + \sum_{j=1}^{n} y_j s_j$$

$$-2d_o \sum_{j=1}^{n} d_j y_j^2 + \left(\frac{d_o^2 - n\gamma}{n^2}\right)\left(\sum_{j=1}^{n} (y_j + s_j)\right)^2$$

$$f_5(y, s) = f_4(y, s) - \left(\frac{\varepsilon}{n^2}\right)\left(\sum_{j=1}^{n} (y_j + s_j)\right)^2$$

Let $P = \{(y, s)\colon\ y \geqq 0,\ s \geqq 0,\ \sum_{j=1}^{n}(y_j + s_j) = n\}$. Consider the following additional decision problems

Problem 6: Is there a $(y, s) \varepsilon P$ satisfying $f_1(y, s) \leqq 0$?

Problem 7: Is there a $(y, s) \varepsilon P$ satisfying $f_2(y, s) \leqq 0$?

Problem 8: Is there a $(y, s) \varepsilon P$ satisfying $f_4(y, s) \leqq 0$?

Problem 9: Is there a $(y, s) \varepsilon P$ satisfying $f_5(y, s) < 0$?

THEOREM 1: Problem 4 is an NP-hard problem.

PROOF: Since $f_1(y, s)$ is a sum of nonnegative terms whenever $(y, s) \varepsilon P$, if $(\bar{y}, \bar{s}) \varepsilon P$ satisfies $f_1(\bar{y}, \bar{s}) \leqq 0$, then we must have $f_1(\bar{y}, \bar{s}) = 0$, this clearly implies from the definition of $f_1(y, s)$, that the following conditions must hold.

$$\sum_{j=1}^{n} d_j \bar{y}_j = d_o,\ \bar{y}_j \bar{s}_j = 0 \text{ and } \bar{y}_j + \bar{s}_j = 1, \text{ for all } j = 1 \text{ to } n$$

These conditions clearly imply that $\bar{y}$ is a solution of the subset sum problem and that the answer to problem 5 is in the affirmative. Conversely if $\hat{y} = (\hat{y}_j)$ is a solution to the subset sum problem, define $\hat{s} = (\hat{s}_j)$ where $\hat{s}_j = 1 - \hat{y}_j$ for each $j = 1$ to n, and it can be verified that $f_1(\hat{y}, \hat{s}) = 0$. This verifies that problems 5 and 6 are equivalent.

Whenever $\bar{y}$ is a 0-1 vector, we have $\bar{y}_j = \bar{y}_j^2$ for all j, and this implies that $f_1(\bar{y}, s) = f_2(\bar{y}, s)$ for any s. So, from the above arguments, we see that if $(\bar{y}, \bar{s}) \varepsilon P$ satisfies $f_1(\bar{y}, \bar{s}) \leqq 0$, then $f_1(\bar{y}, \bar{s}) = f_2(\bar{y}, \bar{s}) = 0$. If $0 \leqq y_j \leqq 1$, we have $2d_0d_jy_j(1 - y_j) \geqq 0$. If $(y, s) \varepsilon P$, and, $y_j > 1$, then $\frac{\gamma}{2}(y_j + s_j - 1)^2 + 2d_0d_jy_j(1 - y_j) \geqq 0$, since γ is large (from the definition of γ). Using this and the definitions of $f_1(y, s)$, $f_2(y, s)$, it can be verified that for $(y, s) \varepsilon P$, if $f_2(y, s) \leqq 0$ then $f_1(y, s) \leqq 0$ too. These facts imply that problems 6 and 7 are equivalent.

Clearly, problems 7 and 8 are equivalent.

From the definition of ε (since it is sufficiently small) and using Lemma 2, one can verify that problems 8 and 9 are equivalent.

Problem 9 is a special case of problem 4. Since problem 5 is NP-complete, from the above chain of arguments we conclude that problem 4 is NP-hard. ■

THEOREM 2: Problem 4 is NP-complete.

PROOF: The answer to problem 4 is in the affirmative iff the answer to the decision problem in the statement of Lemma 1 is in the affirmative. So, from Lemma 1 we conclude that problem 4 is in NP. From Theorem 1, this shows that problem 4 is NP-complete. ■

THEOREM 3: Problem 3 is NP-complete.

PROOF: Problems 3 and 4 are clearly equivalent, this result follows from Theorem 2. ■

THEOREM 4: Both problems 1 and 2 are NP-complete.

PROOF: Problems 1 and 2 are both equivalent to problem 3, so this result follows from Theorem 3. ■

THEOREM 5: Given an integer square matrix D, the decision problem "is D not copositive?" is NP-complete.

PROOF: The decision problem "is D not copositive?" is equivalent to problem 1, hence this result follows from Theorem 4. ■

CAN WE CHECK LOCAL MINIMALITY EFFICIENTLY IN UNCONSTRAINED MINIMIZATION PROBLEMS?

Let $\theta(x)$ be a real valued smooth function defined on R^n. Consider the unconstrained problem

$$\text{minimize } \theta(x). \tag{12}$$

A necessary condition for a given point $\bar{x} \varepsilon R^n$ to be a local minimum for (12) is

$$\nabla\theta(\bar{x}) = 0, \ H(\theta(\bar{x})) \text{ is PSD} \tag{13}$$

where $H(\theta(\bar{x}))$ is the Hessian matrix (the matrix of second order partial derivatives) of $\theta(x)$ at $\bar{x}$. A sufficient condition for $\bar{x}$ to be a local minimum for (12) is

$$\nabla\theta(\bar{x}) = 0, \ H(\theta(\bar{x})) \text{ is positive definite} \tag{14}$$

Both conditions (13) and (14) can be checked very efficiently. If (13) is satisfied, but (14) is violated, there are no simple conditions known to check whether or not $\bar{x}$ is a local minimum for (12). Here, we investigate the complexity of checking whether or not a given point $\bar{x}$ is a local minimum for (12), and that of checking whether $\theta(x)$ is bounded below or not over R^n.

As before, let $D = (d_{ij})$ be an integer square symmetric matrix of order n. Consider the unconstrained problem,

$$\text{minimize } h(u) = (u_1^2, \ldots, u_n^2)D(u_1^2, \ldots, u_n^2)^T \tag{15}$$

Clearly, (15) is an instance of the general unconstrained minimization problem (12). Consider the following decision problems.

Problem 10: Is $\bar{u} = 0$ not a local minimum for (15)?

Problem 11: Is $h(u)$ not bounded below on R^n?

We have, for i, j = 1 to n

$$\frac{\partial h(u)}{\partial u_j} = 4u_j\big((u_1^2, \ldots, u_n^2)D_{.j}\big)$$

$$\frac{\partial^2 h(u)}{\partial u_i \partial u_j} = 8u_i u_j d_{ij}, \; i \neq j$$

$$\frac{\partial^2 h(u)}{\partial u_j^2} = 4(u_1^2, \ldots, u_n^2)D_{.j} + 8u_j^2 d_{jj}$$

where $D_{.j}$ is the j^{th} column vector of D. So, $\bar{u} = 0$ satisfies the necessary conditions for being a local minimum for (15), but not the sufficient condition given in (14).

Using the transformation $x_j = u_j^2$, j = 1 to n, we see that (15) is equivalent to (7). So problems 1 and 10 are equivalent. Likewise, problems 2 and 11 are equivalent. By Theorem 4, we conclude that both problems 10 and 11 are NP-hard. Thus, even in the unconstrained minimization problem, to check whether the objective function is not bounded below, and to check whether a given point is not a local minimum, may be hard problems in general. This also shows that the problem of checking whether a given smooth nonlinear function (even a polynomial) is or is not locally convex at a given point, may be a hard problem in general.

WHAT ARE SUITABLE GOALS FOR ALGORITHMS IN NONCONVEX NLP?

Much of nonlinear programming literature stresses that the goal for algorithms in nonconvex NLPs should be to obtain a local minimum. Our results here show that in general, this may be hard to guarantee.

Many nonlinear programming algorithms are iterative in nature, that is, beginning with an initial point x^o, they obtain a sequence of points $\{x^r: r = 0, 1, ...\}$. For some of the algorithms, under certain conditions, it can be shown that the sequence converges to a KKT point for the original problem, (a KKT point is a feasible solution at which the first order necessary conditions for a local minimum, (4), hold). Unfortunately, there is no guarantee that a KKT point will be a local minimum, and our results point out that in general, checking whether or not it is a local minimum may be a hard problem.

Some algorithms have the property that the sequence of points obtained is actually a descent sequence, that is, either the objective function, or a measure of the infeasibility of the current solution to the problem, or some merit function or criterion function which is a combination of both, strictly decreases along the sequence. Given x^r, these algorithms generate a $y^r \neq 0$ such that the direction $x^r + \lambda y^r$, $\lambda \geq 0$, is a descent direction for the functions discussed above. The next point in the sequence x^{r+1} is usually taken to be the point which minimizes the objective or criterion function on the half-line $\{x^r + \lambda y^r: \lambda \geq 0\}$, obtained by using a line minimization algorithm. On general nonconvex problems, these methods suffer from the same difficulties, they cannot theoretically guarantee that the point obtained at termination is even a local minimum. However, it seems reasonable to expect that a solution obtained through a descent process is more likely to be a local minimum, than a solution obtained purely based on necessary optimality conditions. Thus a suitable goal for algorithms for nonconvex NLPs seems to be a descent sequence converging to a KKT point. Some recently developed algorithms, such as the sequential quadratic programming methods, reach this goal.

References

1. Fiacco, A. V. and G. P. McCormick, **Nonlinear Programming: Sequential Unconstrained Minimization Technique**, Wiley, New York, 1968.

2. Garey, M. R. and D. S. Johnson, **Computers and Intractability, A Guide to the Theory** of **NP-completeness**, Freeman, New York, 1980.

3. Mangasarian, O. L., **Nonlinear Programming**, McGraw-Hill, New York, 1965.

4. Murty, K. G., **Linear Programming**, Wiley, New York, 1983.

5. Murty, K. G., **Linear Complementarity, Linear and Nonlinear Programming**, Heldermann Verlag, West Berlin, 1986, to appear.

6. Murty, K. G. **Linear and Combinatorial Programming**, 1976, available from Krieger Publishing Company, Malabar, Florida.