A Global Optimization Algorithm (GOP) for Certain Classes of Nonconvex NLPs : II. Application of Theory and Test Problems

V. Visweswaran and C.A. Floudas*
Department of Chemical Engineering
Princeton University
Princeton, N.J. 08544-5263

April 1990/Revised : July 1990.

Abstract

In Part I (Floudas and Visweswaran, 1990), a deterministic global optimization approach was proposed for solving certain classes of nonconvex optimization problems. An algorithm, **GOP**, was presented for the rigorous solution of the problem through a series of *primal* and *relaxed dual* problems until the upper and lower bounds from these problems converged to an ϵ-global optimum. In this paper, theoretical results are presented for several classes of mathematical programming problems that include : (i) the general quadratic programming problem, (ii) quadratic programming problems with quadratic constraints, (iii) pooling and blending problems, and (iv) unconstrained and constrained optimization problems with polynomial terms in the objective function and/or constraints. For each class, a few examples are presented illustrating the approach.

Keywords : Global Optimization, Quadratic Programming, Quadratic Constraints, Polynomial functions, Pooling and Blending Problems.

*Author to whom all correspondence should be addressed.

1 Introduction

A large number of nonlinear programming problems can be written in the following form :

$$\begin{aligned} \min_{x,y} \quad & f(x,y) \\ \\ g(x,y) &\leq 0 \\ h(x,y) &= 0 \\ x &\in X \\ y &\in Y \end{aligned} \tag{1}$$

where $f(x,y)$, $g(x,y)$ and $h(x,y)$ can be nonlinear functions leading to nonconvexities in the problem.

Floudas and Visweswaran (1990) proposed a new deterministic global optimization approach for solving problems of the form (1) that satisfy *conditions (A)* (see Part I). The proposed algorithm **(GOP)** uses duality theory to decompose (1) into *primal* and *relaxed dual* subproblems, which are then solved rigorously making use of several theoretical properties utilizing the convexity of the projected problem in the space of the subsets x and y. The algorithm was proved to have finite ϵ-convergence to an ϵ-global optimum of (1).

Through the use of *transformations* and *partitioning* of the variable set, many standard programming problems can be converted to the form given by (1). This paper discusses the mathematical properties of several classes of optimization problems and presents the application of the **GOP** algorithm to some of these classes that have special structure. Of particular interest are problems that commonly appear in chemical engineering applications, such as quadratic programming problems with linear or quadratic constraints, pooling and blending problems, and polynomially constrained problems.

Section 2 discusses the application of the **GOP** algorithm to quadratic programming problems with linear constraints. Examples are presented for bilinear programming and indefinite quadratic programming problems. In section 3, this approach is extended to include quadratic programming problems with linear and quadratic or bilinear constraints. Pooling and blending problems, with linear objective function and bilinear constraints, are considered in section 4. Finally, section 5 considers unconstrained and constrained optimization problems involving polynomial functions in the objective function and/or constraints.

2 The General Quadratic Programming Problem

2.1 Theory

The general quadratic programming problem, with linear constraints, has the following form:

$$\min_{x \in X} \; c^T x + x^T Q x$$

$$\begin{aligned} \textit{subject to} \quad A_1 x - b_1 &\leq 0 \\ A_2 x - b_2 &= 0 \end{aligned}$$

where x an n-vector of variables, and c, b_1 and b_2 are constant vectors. Q, A_1 and A_2 are constant matrices. Depending on the nature of the eigenvalues of the matrix Q, the problem can be a definite or indefinite quadratic programming problem.

By defining a new set of variables $y = x$, and introducing a set of equality constraints, this problem can be converted to the following equivalent problem :

$$\min_{x,y} \; c^T x + x^T Q y$$

$$\begin{aligned} s.t \quad A_1 y - b_1 &\leq 0 \\ A_2 y - b_2 &= 0 \\ x - y &= 0 \end{aligned}$$

By projecting on y, the following linear primal problem is obtained at the k th iteration:

$$\min_{x} \; c^T x + x^T Q y^k$$

$$\begin{aligned} s.t \quad A_1 x - b_1 &\leq 0 \\ A_2 x - b_2 &= 0 \\ x - y^k &= 0 \end{aligned}$$

The solution to this problem provides the optimal multiplier vectors μ^k, λ^k corresponding to the original inequality and equality constraints, and the multipliers ν^k for the equality constraints due to the introduction of the y-variables.

The Lagrange function can be formulated as

$$\begin{aligned} L(x, y, \lambda^k, \mu^k, \nu^k) &= c^T x + x^T Q y + {\mu^k}^T (A_1 x - b_1) + {\lambda^k}^T (A_2 x - b_2) + {\nu^k}^T (x - y) \\ &= x^T Q y + x^T [c + A_1^T \mu^k + A_2^T \lambda^k + \nu^k] - [{\mu^k}^T b_1 + {\lambda^k}^T b_2 + {\nu^k}^T y] \end{aligned} \tag{2}$$

From the KKT gradient condition for the primal problem we have

$$\nabla_x L(x^k, y^k, \lambda^K, \mu^k, \nu^k) = Q y^k + c + A_1^T \mu^k + A_2^T \lambda^k + \nu^k = 0$$

Hence,

$$c + A_1^T \mu^k + A_2^T \lambda^k + \nu^k = -Q y^k \tag{3}$$

Thus, (2) and (3) can be combined to give

$$\begin{aligned} L(x, y, \lambda^k, \mu^k, \nu^k) &= x^T (Q y - Q y^k) - ({\mu^k}^T b_1 + {\lambda^k}^T b_2 + {\nu^k}^T y) \\ &= x^T Q (y - y^k) - ({\mu^k}^T b_1 + {\lambda^k}^T b_2 + {\nu^k}^T y). \end{aligned}$$

Hence, the *qualifying* constraints to be added along with the Lagrange function in the relaxed dual problem take the form

$$\begin{aligned} Q_i (y - y^k) &\leq 0 \quad if \;\; x_i^B = x_i^U \\ Q_i (y - y^k) &\geq 0 \quad if \;\; x_i^B = x_i^L \end{aligned}$$

where Q_i is the i th row of Q, i.e. the row corresponding to x_i in the matrix Q.

For iterations where the primal problem is infeasible, the relaxed primal problem is linear, and hence the Lagrange function generated from this problem (by use of the KKT optimality conditions for the relaxed primal problem) will not contain any terms involving x. Hence, for such iterations, only one *relaxed dual* problem needs to be solved. It should also be noted that if *transformation* variables had been introduced for all the x variables (as in the case of negative definite quadratic problems), the constraints from the original problem can be introduced into the *relaxed dual* problems (in terms of the y variables). For such problems, the primal problem will never be infeasible.

2.2 Computational Results

2.2.1 Bilinear Programming Problems

Example 1 : This example is taken from Konno (1976).

$$\min_{x,y} \quad x_1 - x_2 - y_1 - (x_1 - x_2)(y_1 - y_2)$$

$$\begin{aligned} \textit{subject to} \quad x_1 + 4x_2 &\leq 8 \\ 4x_1 + x_2 &\leq 12 \\ 3x_1 + 4x_2 &\leq 12 \\ 2y_1 + y_2 &\leq 8 \\ y_1 + 2y_2 &\leq 8 \\ y_1 + y_2 &\leq 5 \\ x_1, x_2, y_1, y_2 &\geq 0 \end{aligned}$$

The bilinear terms in the objective function are the only source of nonconvexity in this problem. By *projecting* on y_1 and y_2, the primal and relaxed dual problems become linear in x and y respectively. Since the primal problem is solved for fixed values of y_1 and y_2, the three constraints that contain only y_1 and y_2 can be directly used in the relaxed dual problem.

Iteration 1:

For a starting point of $y_1^1 = 0,\ y_2^1 = 0$, the primal problem is given by

$$\min_{x} \quad x_1 - x_2$$

$$\begin{aligned} \textit{subject to} \quad x_1 + 4x_2 &\leq 8 \\ 4x_1 + x_2 &\leq 12 \\ 3x_1 + 4x_2 &\leq 12 \\ 0 - x_1 &\leq 0 \\ 0 - x_2 &\leq 0 \end{aligned}$$

The solution of this problem yields $x_1 = 0,\ x_2 = 2,\ \mu_1^1 = 0.25,\ \mu_4^1 = 1.25$ and $\mu_2^1 = \mu_3^1 = \mu_5^1 = 0$, where $\mu_1^1, \mu_2^1, \mu_3^1, \mu_4^1$ and μ_5^1 re the Lagrange multipliers for the five constraints. The objective function has a value of -2.

The Lagrange function formulated from this problem is given as

$$\begin{aligned} L^1(x,y,\mu^1) &= x_1 - x_2 - y_1 - (x_1 - x_2)(y_1 - y_2) \\ &\quad +0.25(x_1 + 4x_2 - 8) + 1.25(0 - x_1) \\ &= -(y_1 - y_2)x_1 + (y_1 - y_2)x_2 - y_1 - 2 \end{aligned}$$

on rearranging the terms in x together. From this, it can be seen that the *qualifying* constraints for both x_1 and x_2 are equivalent to $(y_1 - y_2)$ being greater than or less than 0 depending on the nature of the bound chosen for x_1 or x_2. Hence, it is sufficient to solve two *relaxed dual* problems, once for $(y_1 - y_2)$ greater than zero, and once for $(y_1 - y_2)$ being less than zero. From the constraint set, the bounds on x_1 and x_2 are obtained as $x_1^L = x_2^L = 0$, and $x_1^U = 3,\ x_2^U = 2$.

The two *relaxed dual* problems to be solved at the first iteration are shown below:

(i) For $(y_1 - y_2) \geq 0$: The bounds for x are $x_1^B = x_1^U = 3$, and $x_2^B = x_2^L = 0$.

$$\begin{aligned} \min_y \quad & \mu_B \\ s.t. \quad \mu_B \ &\geq \ L^1(x_1 = 3, x_2 = 0, y, \mu^1) = -4y_1 + 3y_2 - 2 \\ & y_1 - y_2 \geq 0 \\ 2y_1 + y_2 \leq 8\ , \quad y_1 + 2y_2 \ &\leq \ 8\ , \quad y_1 + y_2 \leq 5\ , \quad y_1, y_2 \geq 0 \end{aligned}$$

The solution of this problem is $y_1 = 4,\ y_2 = 0$, and $\mu_B = -18$.

(i) For $(y_1 - y_2) < 0$: The bounds for x are $x_1^B = x_1^L = 0$, and $x_2^B = x_2^U = 2$.

$$\begin{aligned} \min_y \quad & \mu_B \\ s.t. \quad \mu_B \ &\geq \ L^1(x_1 = 0, x_2 = 2, y, \mu^1) = y_1 - 2y_2 - 2 \\ & y_1 - y_2 \leq 0 \\ 2y_1 + y_2 \leq 8\ , \quad y_1 + 2y_2 \ &\leq \ 8\ , \quad y_1 + y_2 \leq 5\ , \quad y_1, y_2 \geq 0 \end{aligned}$$

The solution of this problem is $y_1 = 0,\ y_2 = 4$, and $\mu_B = -10$.

Thus, after the first iteration, there are two solutions in the stored set - (i) $\mu_B = -18,\ y = (4,0)$, and (ii) $\mu_B = -10,\ y = (0,4)$. Of these two solutions, the first one gives the lowest bound for the global solution; therefore, this solution is chosen from the stored set. Hence, the lower bound on the problem is -18, and the fixed value of y for the second iteration is (4,0). Since this solution is selected, it is deleted from the stored set.

Iteration 2: For the second iteration, the primal problem is

$$\min_{x} \quad -3x_1 + 3x_2 - 4$$

$$\begin{aligned} \textit{subject to} \quad x_1 + 4x_2 &\leq 8 \\ 4x_1 + x_2 &\leq 12 \\ 3x_1 + 4x_2 &\leq 12 \\ 0 - x_1 &\leq 0 \\ 0 - x_2 &\leq 0 \end{aligned}$$

This problem has the solution $x_1 = 3,\ x_2 = 0, \mu_1^2 = \mu_3^2 = \mu_4^2 = 0,\ \mu_2^2 = 0.75$ and $\mu_5^2 = 3.75$. The objective function value is -13, and becomes the new upper bound on the global solution.

Before solving the two *relaxed dual* problems in the second iteration, a Lagrange function from the first iteration needs to be chosen. To do this, the qualifying constraints for the two Lagrange functions are tested for feasibility at the current y, i.e. at $y = (4, 0)$. At this point,

$$y_1 - y_2 = 4 \geq 0$$

Hence, the Lagrange function that has the qualifying constraint $y_1 - y_2 \geq 0$ is chosen to be present in the *relaxed dual* problems of the current iteration.

The Lagrange function formulated from the second primal problem is

$$\begin{aligned} L^2(x, y, \mu^2) &= x_1 - x_2 - y_1 - (x_1 - x_2)(y_1 - y_2) \\ &\quad +0.75(4x_1 + x_2 - 12) + 3.75(0 - x_2) \\ &= -(y_1 - y_2 - 4)x_1 + (y_1 - y_2 - 4)x_2 - y_1 - 9 \end{aligned}$$

Hence, the qualifying constraints have the form $(y_1 - y_2 - 4)$. The two *relaxed dual* problems for the current iteration are :

(i) For $(y_1 - y_2) \geq 4$: The bounds are $x_1^B = 3$, and $x_2^B = 0$.

$$\begin{aligned} \min_{y} \quad & \mu_B \\ s.t. \quad \mu_B &\geq L^1(x_1 = 3, x_2 = 0, y, \mu^1) = -4y_1 + 3y_2 - 2 \end{aligned}$$

$$
\begin{aligned}
y_1 - y_2 &\geq 0 \\
\mu_B \;\geq\; L^2(x_1 = 3, x_2 = 0, y, \mu^2) &= -4y_1 + 3y_2 + 3 \\
y_1 - y_2 &\geq 4 \\
2y_1 + y_2 \leq 8\,, \quad y_1 + 2y_2 \;\leq\; 8\,, &\quad y_1 + y_2 \leq 5\,, \quad y_1, y_2 \geq 0
\end{aligned}
$$

The solution of this problem is $y_1 = 4,\ y_2 = 0$, and $\mu_B = -13$.

(ii) For $(y_1 - y_2) < 4$: The bounds are $x_1^B = 0$, and $x_2^B = 2$.

$$
\begin{aligned}
\min_{y} \quad & \mu_B \\
s.t. \quad & \mu_B \;\geq\; L^1(x_1 = 3, x_2 = 0, y, \mu^1) = -4y_1 + 3y_2 - 2 \\
& y_1 - y_2 \geq 0 \\
& \mu_B \;\geq\; L^2(x_1 = 0, x_2 = 2, y, \mu^2) = y_1 - 2y_2 - 17 \\
& y_1 - y_2 < 4 \\
& 2y_1 + y_2 \leq 8 \quad y_1 + 2y_2 \;\leq\; 8\,, \quad y_1 + y_2 \leq 5\,, \quad y_1, y_2 \geq 0
\end{aligned}
$$

The solution of this problem is $y_1 = 3.667,\ y_2 = 0.667$, and $\mu_B = -14.667$.

Thus, after the end of the second iteration, there are three solutions in the stored set -

(i) $\mu_B = -10,\ y_1 = 0,\ y_2 = 4$ from the first iteration,

(ii) $\mu_B = -13,\ y_1 = 4,\ y_2 = 0$ from the second iteration, and

(iii) $\mu_B = -14.667,\ y_1 = 3.667,\ y_2 = 0.667$ from the second iteration.

From these, the third solution provides the minimum value of μ_B. Hence, the lower bound is updated to -14.667 and the fixed value of $y = (3.667, 0.667)$ is chosen for the third iteration.

The algorithm continues in this fashion for one more iteration, converging to a solution of -13. This is the global solution for the problem. Similarly, the algorithm took 2-3 iterations to converge from several other starting points.

2.2.2 Indefinite Quadratic Programming

Example 2 :

This example is a large indefinite quadratic programming problem, and is taken from Floudas and Pardalos (1990).

$$\min_{x,y} \quad \Phi(x,y) = \Phi_1(x) + \Phi_2(y)$$

$$\begin{aligned} s.t. \quad A_1 x + A_2 y &\leq b \\ x_i &\geq 0, \quad i = 1, 2...10 \\ y_i &\geq 0, \quad i = 11, 12..20 \end{aligned}$$

where

$$\begin{aligned} \Phi_1(x) &= \frac{\theta_1}{2} \sum_{i=1}^{10} C_i (x_i - \overline{x_1})^2 \quad \theta_1 < 0,\ C_i \geq 0,\ i = 1, 2...10 \\ \Phi_2(y) &= \frac{\theta_2}{2} \sum_{i=11}^{20} C_i (y_i - \overline{y_i})^2 \quad \theta_2 > 0,\ C_i \geq 0,\ i = 11, 12, ...20 \end{aligned}$$

where θ_1 is a negative constant, θ_2 is a positive constant, and C_1, C_2, $\overline{x}$ and $\overline{y}$ are constant vectors. Hence, the function $\Phi_1(x)$ is concave, while $\Phi_2(y)$ is convex. The data for this problem is given below :

$$\begin{aligned} \theta_1 &= -1.00 \quad \theta_2 = 1.00 \\ C &= (63, 15, 44, 91, 45, 50, 89, 58, 86, 82, \\ &\qquad 42, 98, 48, 91, 11, 63, 61, 61, 38, 26) \\ \overline{x} &= (-19, -27, -23, -53, -42, 26, -33, -23, 41, 19) \\ \overline{y} &= (-52, -3, 81, 30, -85, 68, 27, -81, 97, -73) \end{aligned}$$

$$A_1 = \begin{bmatrix} 3 & 5 & 5 & 6 & 4 & 4 & 5 & 6 & 4 & 4 \\ 5 & 4 & 5 & 4 & 1 & 4 & 4 & 2 & 5 & 2 \\ 1 & 5 & 2 & 4 & 7 & 3 & 1 & 5 & 7 & 6 \\ 3 & 2 & 6 & 3 & 2 & 1 & 6 & 1 & 7 & 3 \\ 6 & 6 & 6 & 4 & 5 & 2 & 2 & 4 & 3 & 2 \\ 5 & 5 & 2 & 1 & 3 & 5 & 5 & 7 & 4 & 3 \\ 3 & 6 & 6 & 3 & 1 & 6 & 1 & 6 & 7 & 1 \\ 1 & 2 & 1 & 7 & 8 & 7 & 6 & 5 & 8 & 7 \\ 8 & 5 & 2 & 5 & 3 & 8 & 1 & 3 & 3 & 5 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad A_2 = \begin{bmatrix} 8 & 2 & 4 & 1 & 1 & 1 & 2 & 1 & 7 & 3 \\ 3 & 6 & 1 & 7 & 7 & 5 & 8 & 7 & 2 & 1 \\ 1 & 7 & 2 & 4 & 7 & 5 & 3 & 4 & 1 & 2 \\ 7 & 7 & 8 & 2 & 3 & 4 & 5 & 8 & 1 & 2 \\ 7 & 5 & 3 & 6 & 7 & 5 & 8 & 4 & 6 & 3 \\ 4 & 1 & 7 & 3 & 8 & 3 & 1 & 6 & 2 & 8 \\ 4 & 3 & 1 & 4 & 3 & 6 & 4 & 6 & 5 & 4 \\ 2 & 3 & 5 & 5 & 4 & 5 & 4 & 2 & 2 & 8 \\ 4 & 5 & 5 & 6 & 1 & 7 & 1 & 2 & 2 & 4 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$b = (380, 415, 385, 405, 470, 415, 400, 460, 400, 200)$$

The global solution of this problem occurs at

$$\begin{aligned} \overline{x} &= (0, 0, 0, 62.609, 0, 0, 0, 0, 0, 0) \\ \overline{y} &= (0, 0, 0, 0, 0, 4.348, 0, 0, 0, 0) \end{aligned}$$

In order to remove the nonconvexity in the problem, new transformation variables y_1 through y_{10} are introduced in the following manner:

$$y_1 = x_1,\ \ y_2 = x_2, ..., y_{10} = x_{10}$$

so that

$$\Phi_1(x, y) = \frac{\theta_1}{2} \sum_{i=1}^{10} C_i(x_i - \overline{x_i})(y_i - \overline{y_i})$$

Then, *projecting* on $y_i,\ i = 1, 2, ..., 20$ leads to the primal problem being essentially a function evaluation at $x = y^k$. This implies that the original constraints can be ignored for the primal problem. Hence, the primal problem at the kth iteration can be written as shown below :

$$\min_{x_1, x_2 .. x_{10}} \quad \frac{\theta_1}{2} \sum_{i=1}^{10} C_i(x_i - \overline{x_i})(y_i^k - \overline{y_i}) + \frac{\theta_2}{2} \sum_{i=11}^{20} C_i(y_i^k - \overline{y_i})^2$$

$$subject\ to \quad x_i - y_i^k = 0 \quad i = 1, 2...10$$

The constraints from the original problem are used in the *relaxed dual* problem, since they can be written with $y_i, i = 1, 2, ..., 20$ as the variables (Since y_i is equivalent to x_i for i = 1,2,...,10). It should be noted that if the variables x and y are linearly transformed to account for $\overline{x}$ and $\overline{y}$, this problem is equivalent to the general form of the quadratic programming problem considered in section 2.1, with a diagonal matrix Q formed from the vectors C_1 and C_2, and the constraint matrices being given by the bounding constraints on x and the equality constraints introduced for the concave (x) variables.

Through the formulation of the Lagrange function and use of the KKT conditions for the primal problem, the *qualifying* constraints to be added to the *relaxed dual* problem can be written as

$$\begin{aligned} \theta_1(y_i - y_i^k) &\geq 0 \quad if\ x_i^B = x_i^L, \quad and \\ \theta_1(y_i - y_i^k) &\leq 0 \quad if\ x_i^B = x_i^U \\ & \qquad\qquad for \quad i = 1, 2...10 \end{aligned}$$

The *relaxed dual* problem will thus contain the Lagrange function with x_i set to a combination of bounds, and the corresponding *qualifying* constraint. Also, the original constraints for the problem are present in the problem with the $x_i,\ i = 1, 2, ..., 10$ replaced by the corresponding y_i. This also helps to ensure that the relaxed dual problem never returns infeasible values of y_i for the next iteration.

The **(GOP)** algorithm was applied to this problem from three different starting points, including the starting point of 0. In each case, the globally optimal solution was located in the first iteration, and the algorithm took 3-4 iterations to actually converge to this solution, with 1025 subproblems being solved at every iteration.

3 Quadratic Programming Problems with Quadratic Constraints

3.1 Theory

The quadratic programming problem with quadratic constraints has the following form:

$$\min_{x} \ c^T x + x^T Q x$$

$$\begin{aligned} \textit{subject to} \quad x^T A_m x + Bx - b_m &\le 0 \qquad m = 1, 2 ... p \\ x^T A_m x + Cx - b_m &= 0 \qquad m = p+1, p+2 ... p+q \\ Dx - d &\le 0 \\ Ex - e &= 0 \end{aligned} \tag{4}$$

where x an n-vector of variables, and c, d and e are constant vectors. Q, B, C, D, and E are constant matrices. A_m is an $n \times n$ matrix corresponding to the mth quadratic constraint, and b_m is a constant for that constraint.

By defining a new set of variables $y = x$, this problem can be converted to the following equivalent problem:

$$\min_{x,y} \ c^T x + x^T Q y$$

$$\begin{aligned} \textit{subject to} \quad x^T A_m y + Bx - b_m &\le 0 \qquad m = 1, 2 ... p \\ x^T A_m y + Cx - b_m &= 0 \qquad m = p+1, p+2 ... p+q \end{aligned}$$

$$\begin{aligned} Dx - d &\le 0 \\ Ex - e &= 0 \\ x - y &= 0 \end{aligned}$$

By projecting on y, the following linear primal problem is obtained at the k th iteration:

$$\min_x \; c^T x + x^T Q y^k$$

$$\begin{aligned} \textit{subject to} \quad x^T A_m y^k + Bx - b_m &\le 0 \qquad m = 1, 2 \ldots p \\ x^T A_m y^k + Cx - b_m &= 0 \qquad m = p+1, p+2 \ldots p+q \\ Dx - d &\le 0 \\ Ex - e &= 0 \\ x - y^k &= 0 \end{aligned}$$

The solution to this problem provides the optimal multiplier vectors μ_1^k and λ_1^k corresponding to the quadratic inequality and equality constraints, the multiplier vectors μ_2^k and λ_2^k corresponding to the linear inequality and equality constraints, and the multipliers ν^k for the linear equality constraints due to the introduction of the y-variables.

The Lagrange function can be formulated as

$$\begin{aligned} L(x, y, \lambda_1^k, \mu_1^k, \lambda_2^k, \mu_2^k, \nu^k) &= c^T x + x^T Q y + \sum_{m=1}^{p} \mu_{1_m}^k (x^T A_m y - b_m) + {\mu_1^k}^T Bx \\ &+ \sum_{m=p+1}^{p+q} \lambda_{1_m}^k (x^T A_m y - b_m) + {\lambda_1^k}^T Cx + {\mu_2^k}^T (Dx - d) \qquad (5) \\ &+ {\lambda_2^k}^T (Ex - e) + {\nu^k}^T (x - y) \end{aligned}$$

The KKT gradient condition for this problem is

$$\begin{aligned} \nabla_x L(x^k, y^k, \lambda_1^k, \mu_1^k, \lambda_2^k, \mu_2^k, \nu^k) &= Q y^k + c + \sum_{m=1}^{p} \mu_{1_m}^k A_m y^k + B^T \mu_1^k \\ &\quad + \sum_{m=p+1}^{p+q} \lambda_{1_m}^k A_m y^k + C^T \lambda_1^k + D^T \mu_2^k + E^T \lambda_2^k + \nu^k \\ &= 0 \end{aligned}$$

Hence,

$$c + B^T\mu_1^k + +C^T\lambda_1^k + D^T\mu_2^k + E^T\lambda_2^k + \nu^k = -Qy^k - \sum_{m=1}^{p} \mu_{1_m}^k A_m y^k - \sum_{m=p+1}^{p+q} \lambda_{1_m}^k A_m y^k \qquad (6)$$

Using (5) and (6), the Lagrange function can be reformulated to give

$$\begin{aligned} L(x, y, \lambda_1^k, \mu_1^k, \lambda_2^k, \mu_2^k, \nu) &= x^T(Q + ({\mu_1^k}^T, {\lambda_1^k}^T)A)(y - y^k)^T \\ &\quad -({\mu_1^k}^T, {\lambda_1^k}^T)b - {\mu_2^k}^T d - {\lambda_2^k}^T e - {\nu^k}^T y \end{aligned}$$

Hence, the constraints to be added along with the Lagrange function in the relaxed dual problem take the form

$$\begin{aligned} (Q_i + \sum_{m=1}^{p} \mu_{1_m}^k A_{m_i} + \sum_{m=p+1}^{p+q} \lambda_{1_m}^k A_{m_i}).(y - y^k) &\le 0 \qquad if \quad x_i^B = x_i^U \\ (Q_i + \sum_{m=1}^{p} \mu_{1_m}^k A_{m_i} + \sum_{m=p+1}^{p+q} \lambda_{1_m}^k A_{m_i}).(y - y^k) &\ge 0 \qquad if \quad x_i^B = x_i^L \end{aligned}$$

where the subscript i refers to the ith row of the matrices Q and A_m, i.e., the rows corresponding to the variable x_i in the matrices. Similarly, the *qualifying* constraints for the infeasible primal problem can be generated.

3.2 Computational Results

Example 3 : Hesse's Function

This problem is taken from Hesse (1973). It involves the minimization of a concave function subject to linear and quadratic (nonconvex) constraints. This function has 18 local minima, with a global minimum of -310 at (5,1,5,0,5,10).

$$\min_z \; -25(z_1 - 2)^2 - (z_2 - 2)^2 - (z_3 - 1)^2 - (z_4 - 4)^2 - (z_5 - 1)^2 - (z_6 - 4)^2$$

subject to

$$\begin{aligned} z_1 + z_2 &\ge 2 \\ z_1 + z_2 &\le 6 \end{aligned}$$

$$
\begin{aligned}
-z_1 + z_2 &\leq 2 \\
z_1 - 3z_2 &\leq 2 \\
(z_3 - 3)^2 + z_4 &\geq 4 \\
(z_5 - 3)^2 + z_6 &\geq 4 \\
z_1, z_2 &\geq 0 \\
1 \leq z_3 &\leq 5 \\
0 \leq z_4 &\leq 6 \\
1 \leq z_5 &\leq 5 \\
0 \leq z_6 &\leq 10
\end{aligned}
$$

The following transformations are made :

$$
\begin{aligned}
x_i &= z_i - \overline{z_i} \\
where \quad \overline{z} &= \{2, 2, 1, 4, 1, 4\}
\end{aligned}
$$

Using these transformations, the problem can be written in the form given by (4), with x as the set of variables. The data for this problem is given below :

$$
A_1 = \begin{bmatrix} 0 & & & & & \\ & 0 & & & & \\ & & -1 & & & \\ & & & 0 & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix} \qquad
A_2 = \begin{bmatrix} 0 & & & & & \\ & 0 & & & & \\ & & 0 & & & \\ & & & 0 & & \\ & & & & -1 & \\ & & & & & 0 \end{bmatrix}
$$

$$c^T = (0,0,0,0,0,0)$$

$$B = \begin{bmatrix} 0 & 0 & 4 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4 & -1 \end{bmatrix}$$

$$b = \begin{bmatrix} -4 \\ -4 \end{bmatrix}$$

$$Q = \begin{bmatrix} -25 & & & & & \\ & -1 & & & & \\ & & -1 & & & \\ & & & -1 & & \\ & & & & -1 & \\ & & & & & -1 \end{bmatrix}$$

$$D = \begin{bmatrix} -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -3 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$d^T = (2,2,2,6,2,2,0,4,0,4,4,2,4,6)$$

As the objective function and constraints are nonconvex, it is necessary to introduce new variables y so that the resulting problem **(GOP)** algorithm can be applied. These y variables are defined as :

$$y_i = x_i, \quad i = 1, 2...6$$

From Section 3.1, the *qualifying* constraints to be added along with the Lagrange functions to the *relaxed dual* problem at the kth iteration can be written as :

$$\begin{aligned}
-25 \cdot (y_1 - y_1^k) &\leq 0 \quad if \;\; x_1^B = x_1^U \\
-25 \cdot (y_1 - y_1^k) &\geq 0 \quad if \;\; x_1^B = x_1^L \\
-1 \cdot (y_2 - y_2^k) &\leq 0 \quad if \;\; x_2^B = x_2^U \\
-1 \cdot (y_2 - y_2^k) &\geq 0 \quad if \;\; x_2^B = x_2^L \\
(-1 - \mu_{1_1}^k) \cdot (y_3 - y_3^k) &\leq 0 \quad if \;\; x_3^B = x_3^U \\
(-1 - \mu_{1_1}^k) \cdot (y_3 - y_3^k) &\leq 0 \quad if \;\; x_3^B = x_3^L \\
-1 \cdot (y_4 - y_4^k) &\leq 0 \quad if \;\; x_4^B = x_4^U \\
-1 \cdot (y_4 - y_4^k) &\leq 0 \quad if \;\; x_4^B = x_4^U \\
(-1 - \mu_{1_2}^k) \cdot (y_5 - y_5^k) &\leq 0 \quad if \;\; x_5^B = x_5^U
\end{aligned}$$

$$
\begin{aligned}
(-1-\mu_{1_2}^k)\cdot(y_5-y_5^k) \;&\leq\; 0 \quad if \;\; x_5^B = x_5^U \\
-1\cdot(y_6-y_6^k) \;&\leq\; 0 \quad if \;\; x_6^B = x_6^U \\
-1\cdot(y_6-y_6^k) \;&\leq\; 0 \quad if \;\; x_6^B = x_6^U
\end{aligned}
$$

The **(GOP)** algorithm was applied to this problem from several starting points. In each case, the optimal solution of -310 was found in 3-4 iterations, with 65 subproblems being solved at every iteration .

Example 4 :

This example features the minimization of a linear function of two variables subject to two quadratic constraints, one of which is nonconvex, and two linear constraints.

$$
\min_{y} \quad y_1 + y_2
$$

$$
\begin{aligned}
s.t. \quad y_1^2 + y_2^2 \;&\leq\; 4 \\
y_1^2 + y_2^2 \;&\geq\; 1 \\
y_1 - y_2 - 1 \;&\leq\; 0 \\
y_2 - y_1 - 1 \;&\leq\; 0 \\
-2 \leq y_1 \;&\leq\; 2 \\
-2 \leq y_2 \;&\leq\; 2
\end{aligned}
$$

The feasible region formed by the constraint set is shown in Figure 1. As can be seen, there are two distinct but separated regions where the problem is feasible. Hence, a conventional solver cannot be expected to determine the global minimum of -2.828 at $(-1.414, -1.414)$.

If the following two variables are introduced :

$$
x_1 = y_1 \;, \quad and \;\; x_2 = y_2
$$

the problem can be written in the following equivalent form :

$$
\min_{x,y} \quad x_1 + x_2
$$

$$
\begin{aligned}
s.t. \quad x_1y_1 + x_2y_2 - 4 \;&\leq\; 0 \\
1 - x_1y_1 - x_2y_2 \;&\leq\; 0
\end{aligned}
$$

$$\begin{aligned} x_1 - x_2 - 1 &\leq 0 \\ -x_1 + x_2 - 1 &\leq 0 \\ x_1 - y_1 &= 0 \\ x_2 - y_2 &= 0 \end{aligned}$$

Iteration 1 :

Consider a starting point of (1,1) for the **GOP** algorithm. The solution of the primal problem, which is feasible, yields $x_1 = 1,\ x_2 = 1$ and the multipliers $\mu^1_{1_1} = \mu^1_{1_2} = \mu^1_{2_1} = \mu^1_{2_2} = 0,\ \nu^1_1 = \nu^1_2 = -1$. The objective function has a value of 2.

The Lagrange function formulated from the first primal problem is

$$\begin{aligned} L^1(x, y, \mu^1_1, \mu^1_2, \nu^1) &= x_1 + x_2 - 1 \cdot (x_1 - y_1) - 1 \cdot (x_2 - y_2) \\ &= y_1 + y_2 \end{aligned}$$

This Lagrange function is independent of x_1 and x_2. Hence, the *relaxed dual* problem needs to be solved only once, since it does not depend on the bounds of the x variables. This *relaxed dual* problem is given below :

$$\min_{y, \mu_B} \quad \mu_B$$

$$\begin{aligned} s.t. \quad \mu_B &\geq y_1 + y_2 \\ y_1 - y_2 - 1 &\leq 0 \\ -y_1 + y_2 - 1 &\leq 0 \\ -2 \leq y_1 &\leq 2 \\ -2 \leq y_2 &\leq 2 \end{aligned}$$

where the linear inequality constraints from the original problem have been added in terms of y_1 and y_2. The solution of this problem is $y = (-2, -2)$, and $\mu_B = -4$. Since this is the only relaxed dual problem solved at the first iteration, there is only one stored value of (μ_B, y). Hence, the fixed value of y for the second iteration is $(-2, -2)$, and the lower bound for the optimal solution is -4.

Iteration 2 :

For the second iteration, for a fixed value of $y = (-2, -2)$, the primal problem is infeasible because the first quadratic constraint is violated. Hence, a *relaxed* primal problem is solved.

A form of this problem, with the slack variables α_1 and α_2 only in the nonlinear constraints, is shown below :

$$\min_{x,\alpha} \quad \alpha_1 + \alpha_2$$

$$\begin{aligned} s.t. \qquad -2x_1 - 2x_2 - 4 - \alpha_1 &\leq 0 \\ 1 + 2x_1 + 2x_2 - \alpha_2 &\leq 0 \\ x_1 - x_2 - 1 &\leq 0 \\ -x_1 + x_2 - 1 &\leq 0 \\ x_1 + 2 &= 0 \\ x_2 + 2 &= 0 \end{aligned}$$

The solution of this problem is $x = (-2, -2)$, $\alpha_1 = 4$ and $\alpha_2 = 0$. The optimal multipliers for this problem are $\mu^2_{1_1} = 1$, $\nu^2_1 = \nu^2_2 = 2$, and $\mu^2_{1_2} = \mu^2_{2_1} = \mu^2_{2_2} = 0$. The sum of the infeasibilities is 4.

The Lagrange function from this problem can be formulated as

$$\begin{aligned} L^2(x, y, \mu^2_1, \mu^2_2, \nu^2) &= 1 \cdot (x_1y_1 + x_2y_2 - 4) + 2 \cdot (x_1 - y_1) + 2 \cdot (x_2 - y_2) \\ &= (y_1 + 2)x_1 + (y_2 + 2)x_2 - 2y_1 - 2y_2 - 4 \end{aligned}$$

where the terms containing α_1 and α_2 vanish because of the KKT conditions for the second primal problem. The Lagrange function to be added to the *relaxed dual* problem for the second iteration has the form

$$L^2(x, y, \mu^2_1, \mu^2_2, \nu^2) \leq 0$$

The *qualifying* constraints of this Lagrange function w.r.t x_1 and x_2 are given as

$$\begin{aligned} \nabla_{x_1} L^2 = y_1 + 2 &\geq 0 \quad if\ x^B_1 = x^L_1 = -2 \\ &\leq 0 \quad if\ x^B_1 = x^U_1 = 2 \\ \nabla_{x_2} L^2 = y_2 + 2 &\geq 0 \quad if\ x^B_2 = x^L_2 = -2 \\ &\leq 0 \quad if\ x^B_2 = x^U_2 = 2 \end{aligned}$$

There are four *relaxed dual* problems solved at the second iteration, corresponding to the combinations of bounds $(2, 2)$, $(-2, 2)$, $(2, -2)$, and $(-2, -2)$ for x_1 and x_2. Of these, the first is irrelevant, since this corresponds to $y_1 \leq -2$ and $y_2 \leq -2$, which implies that this problem cannot provide any useful solution. Also, the *relaxed dual* problems corresponding to $(-2,2)$ and $(2,-2)$ for x^B are complementary due to the symmetry of the problem. Hence, only one of these problems, say, for $x^B = (-2,2)$, needs to be solved, along with the *relaxed dual* problems corresponding to $x^B = (-2,-2)$.

There is one Lagrange function from the first iteration, and it is present since it depends only on y (i.e., its *qualifying* constraints are always satisfied). The *relaxed dual* problem solved with $x_B = (-2,2)$ has a solution of $y = (-1,-2)$ and $\mu_B = -4$. The complementary *relaxed dual* problem, that is, the relaxed dual problem solved with ($x^B = (2,-2)$), has the solution $y = (-2,-1)$, with $\mu_B = -4$. The *relaxed dual* problem solved with $x_B = (-2,-2)$ has a solution of $y = (-1,-2)$ and $\mu_B = -4$, which is identical to the one found for the *relaxed dual* problem solved with $x^B = (-2,2)$.

After the second iteration, there are two stored solutions to choose from :

$$\mu_B = -4 \ , \quad y = (-1,-2) \quad and \quad \mu_B = -4 \ , \quad y = (-2,-1)$$

Since both these solutions have the same value for μ_B, either one can be chosen. Suppose the first solution is chosen. Then, the fixed value of y for the third iteration is $(-1,-2)$. The lower bound for the optimal solution remains at -4.

Iteration 3 :

When the third *relaxed* primal problem is solved (since (-1,-2) is infeasible for the primal problem), the solution is $x = (-1,-2)$, $\mu^3_{1_1} = 1$, $\nu^3_1 = 1$, $\nu^3_2 = 2$, and $\mu^3_{1_2} = \mu^3_{2_1} = \mu^3_{2_2} = 0$.

Before solving the relaxed dual problems, the *qualifying* constraints for the Lagrange function from the second iteration need to be checked at $y = (-1,-2)$. This results in

$$\begin{aligned} \nabla_{x_1} L^2(y_1 = -1, y_2 = -2) &= -1 + 2 \geq 0 \\ \nabla_{x_2} L^2(y_1 = -1, y_2 = -2) &= -2 + 2 = 0 \end{aligned}$$

Thus, from the second iteration, the Lagrange functions corresponding to $x^B = (-2,-2)$ and $x^B = (-2,2)$ can be present in the current relaxed dual problems. However, if both these Lagrange functions are selected, it is equivalent to fixing the value of y_2 at -2 for the current relaxed dual problems, which leads to no improvement in the solutions or infeasible solutions.

This difficulty can be avoided by introducing the *qualifying* constraints in a perturbed form into the *relaxed dual* problems (see Section 6.2 of Floudas and Visweswaran, 1990). This ensures that only one Lagrange function from every previous iteration can be selected.

The **GOP** algorithm was applied to the problem in this form, and the global solution of -2.828 was identified. The algorithm took 127 iterations to converge, solving 5 subproblems at every iteration.

<u>**Note :**</u> The constraint $y_1^2 + y_2^2 \leq 4$ is a convex constraint, and hence can be kept unaltered in the problem formulation. If this is done, with only the nonconvex constraint being transformed, the **GOP** algorithm converges to the optimal solution in two iterations.

4 Pooling and Blending Problems

4.1 Theory

Pooling and blending problems are a feature of the models of most chemical processes. In particular, for problems relating to refinery and petrochemical processing, it is often necessary to model not only the product flows but the properties of intermediate streams as well. These streams are usually combined in a tank or pool, and the pool is used in downstream processing or blending. The presence of these streams in the model introduces nonlinearities, often in a nonconvex manner.

A general formulation of the pooling or blending problem is given below:

$$\min_{x,p} \quad c^T x$$

$$\begin{aligned} \textit{subject to} \quad pA_1x + B_1x - b_1 &= 0 \\ pA_2x + B_2x - b_2 &\leq 0 \end{aligned}$$

where x is an n-vector, A_1 and B_1 are $m_1 \times n$ matrices, corresponding to m_1 equality constraints, and A_2 and B_2 are $m_2 \times n$ matrices corresponding to m_2 inequality constraints. The vectors b_1, b_2 and c are constant vectors, and p is a *scalar variable* representing the pool (or tankage) quality.

The nonconvexities in this problem come from the presence of the bilinear terms in the equality constraints. By selecting the pool quality p to be the complicating variable, the primal problem becomes linear in x. At the kth iteration, the primal problem, for a fixed value p^k of the complicating variable, is given by

$$\min_{x} \quad c^T x$$

$$\begin{aligned} subject\ to \quad p^K A_1 x + B_1 x - b_1 &= 0 \\ p^K A_2 x + B_2 x - b_2 &\le 0 \end{aligned}$$

The Lagrange function for this problem is

$$\begin{aligned} L(x, p, \lambda^K, \mu^K) &= c^T x + {\lambda^K}^T (pA_1 x + B_1 x - b_1) + {\mu^K}^T (pA_2 x + B_2 x - b_2) \\ &= x^T c + x^T A_1^T \lambda^K p + x^T B_1^T \lambda^K - {\lambda^K}^T b_1 \\ &\quad + x^T A_2^T \mu^K p + x^T B_2^T \mu^K - {\mu^K}^T b_2 \end{aligned}$$

Collecting the terms in x together leads to the following:

$$L(x, p, \lambda^K, \mu^K) = x^T (c + A_1^T \lambda^K p + B_1^T \lambda^K + A_2^T \mu^K p + B_2^T \mu^K) - {\lambda^K}^T b_1 - {\mu^K}^T b_2 \quad (7)$$

The KKT gradient conditions for the Kth primal problem are given by

$$c + A_1^T \lambda^K p^K + B_1^T \lambda^K + A_2^T \mu^K p^K + B_2^T \mu^K = 0 \quad (8)$$

From (7) and (8), the Lagrange function can be written as

$$L(x, p, \lambda^K, \mu^K) = x^T (A_1^T \lambda^K p + A_2^T \mu^K p - A_1 \lambda^K p^K - A_2 \mu^K p^K) - \lambda^T b_1 - \mu^T b_2$$

and, since p is a scalar variable,

$$L(x, p, \lambda^K, \mu^K) = x^T (A_1^T \lambda^K + A_2^T \mu^K).(p - p^K) - \lambda^T b_1 - \mu^T b_2$$

Hence, at the Kth iteration, we can consider the relaxed dual problem separately over the two regions of p given by $p - p^K \le 0$ and $p - p^K \ge 0$. For each of these regions, the appropriate bounds of x_i can be found as follows:

(i) For $p - p^K \ge 0$

$$\begin{aligned} &If \quad (A_{1_i}^T \lambda^K + A_{2_i}^T \mu^K) \ge 0 \quad then \quad x_i^B = x_i^L \\ &If \quad (A_{1_i}^T \lambda^K + A_{2_i}^T \mu^K) \le 0 \quad then \quad x_i^B = x_i^U \end{aligned} \quad (9)$$

(i) For $p - p^K \le 0$

$$\begin{aligned} &If \quad (A_{1_i}^T \lambda^K + A_{2_i}^T \mu^K) \ge 0 \quad then \quad x_i^B = x_i^U \\ &If \quad (A_{1_i}^T \lambda^K + A_{2_i}^T \mu^K) \le 0 \quad then \quad x_i^B = x_i^L \end{aligned}$$

where A_{1_i}, A_{2_i} are the ith columns of A_1 and A_2 respectively.

Hence, at each iteration, the relaxed dual problem needs to be solved once for each region of p (less than or greater than p^K respectively), with the noncomplicating variables x_i set to the appropriate bound according to the criteria given by (9).

4.2 Computational Results

Example 5 : Haverly's Pooling Problem

In his studies of the recursive behaviour of Linear Programming **(LP)** models, Haverly (1978) defined a **pooling problem** as shown in Figure 2. Three substances A, B and C with different sulfur contents are to be combined to form two products x and y with specified maximum sulfur contents. In the absence of a pooling restriction, the problem can be formulated and solved as a **LP**. However, when the streams need to be pooled (as, for example, when there is only one tank to store A and B), the **LP** must be modified. Haverly showed that without the explicit incorporation of the effect of the economics associated with the sulfur constraints on the feed selection process, a recursive algorithm for solving a simple formulation having only a pool balance cannot find the global solution. Lasdon *et al.* (1979) added a pool quality constraint to the formulation. This complete nonlinear programming **NLP** formulation is shown below:

$$min \quad Cost = 6A + 16B + 10(Cx + Cy) - 9x - 15y$$

$$s.t. \qquad Px + Py - A - B = 0 \;\Big\} \qquad \textit{pool balance}$$

$$\left.\begin{array}{rcl} x - Px - Cx & = & 0 \\ y - Py - Cy & = & 0 \end{array}\right\} \qquad \textit{component balance}$$

$$p.(Px + Py) - 3A - B = 0 \;\Big\} \qquad \textit{pool quality}$$

$$\left.\begin{array}{rcl} p.Px + 2.Cx - 2.5x & \leq & 0 \\ p.Py + 2.Cy - 1.5y & \leq & 0 \end{array}\right\} \qquad \textit{product quality constraints}$$

$$\left.\begin{array}{rcl} x & \leq & x^U \\ y & \leq & y^U \end{array}\right\} \qquad \textit{upper bounds on products}$$

where p is the sulfur quality of the pool; its lower and upper bounds are 1 and 3 respectively. Lasdon solved this **NLP** using two different procedures, but in each case, the optimal solution found depended on the starting point.

Haverly (1979) suggested a formulation using **correction vectors** representing the pool quality over and under that assumed for the pooling coefficient. This was solved by both Haverly (1979) and Lasdon *et al.* (1979). In both cases, however, the global optimum could not always be found, the solution being dependent on the starting point.

More recently, Floudas and Aggarwal (1990) solved this problem using the Global Optimum Search (Floudas *et al*, 1989). They had to reformulate the problem by adding variables and constraints, and even though they were successful in finding the global minimum from 28 out of 30 starting points, they could not mathematically guarantee that the algorithm would converge to the global minimum.

The GOP Algorithm :

The Haverly pooling problem can be represented in the form given in section 4.1 with the following data :

$$n = 8, \qquad m_1 = 4, \qquad m_2 = 3$$

$$x = (A, B, Cx, Cy, Px, Py, x, y) \qquad b_1^T = (0,0,0,0)$$

$$c^T = (6, 16, 10, 10, 0, 0, -9, -15) \qquad b_2^T = (0,0)$$

$$A_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix} \qquad B_1 = \begin{bmatrix} -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 & 0 & 1 \\ -3 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \qquad B_2 = \begin{bmatrix} 0 & 0 & 2 & 0 & 0 & 0 & -2.5 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & -1.5 \end{bmatrix}$$

By *projecting* on p, the pooling quality, the problem becomes linear in the remaining variables.

From the matrices A_1 and A_2, it can be seen that only Px and Py are the *connected* variables. That is, the terms containing the remaining "x" variables (namely, A, B, Cx, Cy, x and y) in the Lagrange function added to the **RD** problem will disappear. Hence, only the variables Px and Py need to be set to the appropriate bounds for each **RD** problem. Using (9), these bounds can be found as follows :

(i) For $p \geq p^K$:

$$\begin{array}{llr} Px^B = Px^L & if & (\lambda_4^K + \mu_1^K) \geq 0 \\ Px^B = Px^U & if & (\lambda_4^K + \mu_1^K) \leq 0 \\ Py^B = Py^L & if & (\lambda_4^K + \mu_2^K) \geq 0 \\ Py^B = Py^U & if & (\lambda_4^K + \mu_2^K) \leq 0 \end{array}$$

(ii) For $p \leq p^K$: The opposite of the bounds given above.

Similarly, the *qualifying* constraints to be added to the relaxed dual problem at the Kth iteration along with the Lagrange function can be derived from (9).

Three cases of the Pooling problem were solved using the **GOP** algorithm. In the first case, the upper bounds of x and y are 100 and 200 respectively. The global minimum for this problem lies at $p = 1$, with the objective function being equal to -\$400. For the second case, the upper bound of x is changed to 600, while the upper bound of y is 200. This case of the problem has a global optimum of -\$600 occurring at $p = 3$. In the third case,the upper bounds on x and y are the same as for first case, but the cost coefficient of B in the objective function is changed from \$16 to \$13. In this case, the problem attains its global minimum at $p = 1.5$, with the objective function being -\$750.

The **GOP** algorithm was applied to this problem from several starting points. The algorithm found the global optimum in each case from all the starting points. The results for Case I, Case II and Case III are shown in Table 1. For Cases I and II, the algorithm required an average of 15 iterations to converge, while it took only 6-7 iterations for the third case.

Example 6 : Multiperiod Tankage Quality Problem

This example concerns the application of the **(GOP)** algorithm to a multiperiod tankage quality problem. The following sets are defined for the mathematical formulation of the problem :

$$\begin{array}{rcll} PR & = & \{p\} & : \text{ set of products} \\ CO & = & \{c\} & : \text{ set of components} \\ T & = & \{t\} & : \text{ set of timeperiods} \\ QL & = & \{l\} & : \text{ set of qualities} \end{array}$$

Starting Point	Optimal Solution Found		
(p)	Case I	Case II	Case III
1.00	-400	-600	-750
1.25	-400	-600	-750
1.50	-400	-600	-750
1.75	-400	-600	-750
2.00	-400	-600	-750
2.25	-400	-600	-750
2.50	-400	-600	-750
2.75	-400	-600	-750
3.00	-400	-600	-750

Table 1: Results for the Pooling Problem

For this problem, there are 3 products $(p1, p2, p3)$, 2 components $(c1, c2)$, and 3 time periods $(t0, t1, t2$ - includes the time period corresponding to the starting point). The following variables are defined :

$$x_{c,p,t} \;:\; \text{amount of component } c \text{ allocated to product } p \text{ at period } t$$

$$s_{p,t} \;:\; \text{stock of product } p \text{ at end ofperiod } t$$

$$q_{p,l,t} \;:\; \text{quality } l \text{ of product } p \text{ at period } t$$

The objective of the problem is to maximize the total value at the end of the last time period. The terminal value of each product (val_p) is given. Lower and upper bounds on the quality variables are provided as well as initial quality values. Limits on product stocks ($stock_{p,t}$) for each time period are also provided. Data for the qualities in each component $(QU_{c,l})$ and the product lifting $(LF_{p,t})$ is also provided for the problem.

The complete mathematical formulation for this problem, consisting of 39 variables and 22 inequality constraints (of which 12 are nonconvex) is given below :

$$\max \quad \sum_{p \in PR} val_p . s_{p,'t2'}$$

subject to

$$\sum_{p\in PR} x_{c,p,t} \le AR_{c,t} \qquad t\in\{t1,t2\},\ c\in CO$$
$$s_{p,t}+\sum_{c\in CO} x_{c,p,t+1}-s_{p,t+1} \ge LF_{p,t+1} \qquad t\in\{t0,t1\},\ p\in PR$$
$$s_{p,t}.q_{p,l,t}+\sum_{c\in CO} x_{c,p,t+1}.QU_{c,l} \ge$$
$$(s_{p,t+1}+LF_{p,t+1}).q_{p,l,t+1} \qquad t\in\{t0,t1\},\ p\in PR,\ l\in QL$$

The sources of nonconvexities in this problem are the bilinear terms $s_{p,t}.q_{p,l,t}$ in the last set of constraints. Thus, fixing either the set of s or q variables makes the problem linear in the remaining variables. Here, the q variables are chosen to be the y variables, i.e. they are fixed for the primal problem. Then, the variables s are the *connected* variables.

For a fixed $q=q^k$, the primal problem is given by :

$$\min \quad \sum_{p\in PR} -val_p.s_{p,'t2'}$$

subject to

$$\sum_{p\in PR} x_{c,p,t} \le AR_{c,t} \qquad t\in\{t1,t2\},\ c\in CO \tag{10}$$
$$-s_{p,t}-\sum_{c\in CO} x_{c,p,t+1}+s_{p,t+1} \le -LF_{p,t+1} \qquad t\in\{t0,t1\},\ p\in PR \tag{11}$$
$$-s_{p,t}.q^k_{p,l,t}-\sum_{c\in CO} x_{c,p,t+1}.QU_{c,l} \le$$
$$-(s_{p,t+1}+LF_{p,t+1}).q^k_{p,l,t+1} \qquad t\in\{t0,t1\},\ p\in PR,\ l\in QL \tag{12}$$
$$0-s_{p,t} \le 0 \qquad p\in PR, t\in\{t0,t1\} \tag{13}$$
$$s_{p,t}-stock_{p,t} \le 0 \qquad p\in PR, t\in\{t0,t1\} \tag{14}$$

where the bounds on the stocks s have been explicitly incorporated in the problem. The problem has been written as a minimization problem by multiplying the objective function by -1. It should be noted that $s(p,'t0')$, the stock of product p at the beginning of the first period, is fixed.

The KKT gradient conditions for the x variables are given as

$$\nabla_{x_{c,p,t}} L(x,s,q^k,\mu^k) = \mu^k_{1_{c,t}} - \mu_{2_{p,t-1}} - \sum_{l\in QL} \mu_{3_{p,l,t-1}}.QU_{c,l} = 0$$

The KKT gradient conditions for the *connected* variables s in this problem are given by

(i) For $t =' t1',\ p \in PR$,

$$\nabla_{s_{p,t}} L(x, s, q^k, \mu^k) = \mu^k_{2_{p,t-1}} - \mu^k_{2_{p,t}} + \sum_{l \in QL} (\mu_{3_{p,l,t-1}} - \mu_{3_{p,l,t}}).q^k_{p,l,t} - \mu^k_{L_{p,t}} + \mu^k_{U_{p,t}} = 0$$

(ii) For $t =' t2',\ p \in PR$,

$$\nabla_{s_{p,t}} L(x, s, q^k, \mu^k) = -val_p + \mu^k_{2_{p,t-1}} - \mu^k_{2_{p,t}} + \sum_{l \in QL} (\mu_{3_{p,l,t-1}} - \mu_{3_{p,l,t}}).q^k_{p,l,t} - \mu^k_{4_{p,t}} + \mu^k_{5_{p,t}} = 0$$

where $\mu^k_1, \mu^k_2, \mu^k_3, \mu^k_4$ and μ^k_5 correspond to the constraint sets (1) to (5) respectively.
The Lagrange function formulated from the kth primal problem is given by

$$\begin{aligned} L(x, s, q, \mu^k) \ = \ & \sum_{p \in PR} -val_p.s_{p,'t2'} + \sum_{\substack{t \in \{t1,t2\} \\ c \in CO}} \mu_{1_{c,t}} \Big(\sum_{p \in PR} x_{c,p,t} - AR_{c,t} \Big) \\ & + \sum_{\substack{t \in \{t0,t1\} \\ p \in PR}} \mu^k_{2_{p,t}} \Big(-s_{p,t} - \sum_{c \in CO} x_{c,p,t+1} + s_{p,t+1} + LF_{p,t+1} \Big) \\ & + \sum_{\substack{t \in \{t0,t1\} \\ p \in PR \\ l \in QL}} \mu_{3_{p,l,t}} \Big(-s_{p,t}.q_{p,l,t} - \sum_{c \in CO} x_{c,p,t+1}.QU_{c,l} + [s_{p,t+1} + LF_{p,t+1}].q_{p,l,t+1} \Big) \\ & + \sum_{\substack{t \in \{t1,t2\} \\ p \in PR}} \mu_{4_{p,t}} (0 - s_{p,t}) + \sum_{\substack{t \in \{t1,t2\} \\ p \in PR}} \mu_{5_{p,t}} (s_{p,t} - stock_{p,t}) \end{aligned}$$

Using the KKT gradient conditions for the x variables, it can be seen that the terms in x in the Lagrange function will vanish (due to the fact that they are not *connected* variables). This, along with the use of the KKT gradient conditions for the s variables, enables the Lagrange function to be written in the following form :

$$\begin{aligned} L(x, s, q, \mu^k) \ = \ & - \sum_{\substack{t \in \{t1,t2\} \\ c \in CO}} \mu_{1_{c,t}} - AR_{c,t} + \sum_{\substack{t \in \{t0,t1\} \\ p \in PR}} \mu^k_{2_{p,t}}.LF_{p,t+1} \\ & + \sum_{\substack{t \in \{t0,t1\} \\ p \in PR \\ l \in QL}} \mu_{3_{p,l,t}}.LF_{p,t+1}.q_{p,l,t+1} - \sum_{\substack{t \in \{t1,t2\} \\ p \in PR}} \mu_{5_{p,t}}.stock_{p,t} \\ & + \sum_{\substack{t \in \{t1,t2\} \\ p \in PR \\ l \in QL}} (\mu^k_{3_{p,l,t-1}} - \mu^k_{3_{p,l,t}}).(q_{p,l,t} - q^k_{p,l,t}).s_{p,t} \end{aligned}$$

Thus, the *qualifying* constraints to be added along with the Lagrange function to the *relaxed dual* problem are of the form

$$\sum_{l \in QL} (\mu^k_{3_{p,l,t-1}} - \mu^k_{3_{p,l,t}}).(q_{p,l,t} - q^k_{p,l,t}) \geq 0 \quad if \ s_{p,t} = s^L_{p,t}$$
$$\sum_{l \in QL} (\mu^k_{3_{p,l,t-1}} - \mu^k_{3_{p,l,t}}).(q_{p,l,t} - q^k_{p,l,t}) < 0 \quad if \ s_{p,t} = s^U_{p,t}$$

for all $t \in \{t1, t2\}, \ p \in PR$.

There are six s variables (corresponding to three products at two time periods $t1$ and $t2$). Hence, there are 64 *relaxed dual* problems solved at every iteration.

The **(GOP)** algorithm was applied to the problem in this form. It found the global solution of -9.5316 from all considered starting points. Starting from the lower bound, the algorithm took 16 iterations to converge, solving 65 subproblems in every iteration.

5 Optimization Problems with Polynomial Functions in the Objective and/or Constraints

5.1 Theory

In this section, the application of the **GOP** algorithm to optimization problems involving polynomial functions in the objective function and/or constraints is presented. It is assumed for simplicity that the problem is restricted to cases involving one variable. However, the approach can be extended to problems with more than one variable appearing polynomially in the objective function and constraints.

Consider the following general problem :

$$\min_{y \in Y} \quad f(y) = a_0 + a_1 y + a_2 y^2 + ... + a_s y^s$$

$$A_{j0} + A_{j1} y + A_{j2} y^2 + ... + A_{js} y^s \ \leq \ 0 \quad \forall \ j = 1, 2...m \qquad (15)$$
$$B_{j0} + B_{j1} y + B_{i2} y^2 + ... + B_{js} y^s \ = \ 0 \quad \forall \ j = 1, 2...n$$

where y is a *single* variable and A_{ji} is the coefficient of y^i in the jth constraint. This problem can be nonconvex due to the existence of polynomial terms in the either the objective function or the set of constraints. It is assumed that the polynomial has nonconvex terms right up to the s^{th} degree term.

Consider the following transformations:

$$\begin{array}{rclcl} x_0 & = & 1 & & \\ x_1 & = & y & & \\ x_2 & = & y^2 & = & x_1 y \\ x_3 & = & y^3 & = & x_2 y \\ \vdots & & \vdots & & \vdots \\ x_s & = & y^s & = & x_{s-1} y \end{array}$$

Hence, the primal problem, for a fixed $y = y^K$, can be written as

$$\min_x \sum_{i=0}^{s} a_i x^i$$

$$\begin{array}{rcll} \sum_{i=0}^{s} A_{ji} x_i & \leq & 0 & j = 1, 2...m \\ \sum_{i=0}^{s} B_{ji} x_i & = & 0 & j = 1, 2...n \\ x_i - x_{i-1} y^K & = & 0 & i = 1, 2..s \end{array}$$

where $x_0 = 1$. The KKT conditions for this problem are

$$\nabla_{x_i} L(x, y, \lambda^K, \mu^K, \nu^K) = a_i + \sum_{j=1}^{m} \mu_j^K A_{ji} + \sum_{j=1}^{n} \lambda_j^K B_{ji} + \nu_i^K - \nu_{i+1}^K y^K = 0 \quad \forall\ i = 1, 2...s$$

where λ^K and μ^K correspond to the original equality and inequality constraints, and ν^K correspond to the new equality constraints introduced, with $\nu_0^K = \nu_{s+1}^K = 0$.

The Lagrange function for this problem is given by

$$L(x, y, \lambda^K, \mu^K, \nu^K) = \sum_{i=0}^{s} a_i x_i + + \sum_{j=1}^{m} \mu_j^K (\sum_{i=1}^{s} A_{ji} x_i) + \sum_{j=1}^{n} \lambda_j^K (\sum_{i=1}^{s} B_{ji} x_i) + \sum_{i=1}^{s} \nu_i^K (x_i - x_{i-1} y)$$

Separating the terms in x, this can also be written as

$$L(x, y, \lambda^K, \mu^K, \nu^K) = \sum_{i=0}^{s} \left[a_i + \sum_{j=1}^{m} A_{ji} \mu_j^K + \sum_{j=1}^{n} B_{ji} \lambda_j^K + \nu_i^K - \nu_{i+1}^K y \right] x_i \qquad (16)$$

Using the KKT conditions, the Lagrange function can be written as

$$L(x, y, \lambda^K, \mu^K, \nu^K) = \sum_{i=0}^{s} \nu_{i+1}^K (y^K - y) x_i$$

Thus, the *qualifying* constraint for all the x_i are of the form

$$y^K - y \leq 0, \ \ or \quad y^K - y \geq 0$$

It is therefore sufficient to solve the relaxed dual for these two regions of y, with x_i set to the appropriate bounds. The bounds for x_i for these two relaxed dual problems can be selected as follows:

(i) For $y^K - y \geq 0$

$$\begin{aligned} If \quad \lambda_{i+1}^K &\geq 0, \quad then \quad x_i = x_i^L \\ \lambda_{i+1}^K &\leq 0, \quad then \quad x_i = x_i^U \end{aligned}$$

(ii) For $y^K - y \leq 0$

$$\begin{aligned} If \quad \lambda_{i+1}^K &\geq 0, \quad then \quad x_i = x_i^U \\ \lambda_{i+1}^K &\leq 0, \quad then \quad x_i = x_i^L \end{aligned}$$

Using these combinations of bounds, the two relaxed dual problems can be solved for the appropriate regions of y.

This approach has been outlined for cases when the problem has only one variable. But as the following example shows, the presence of other variables, as long as they do not contribute to nonconvexities in the problem, does not affect the procedure outlined above. This is because such variables can be treated as *nonconnected* variables.

5.2 Computational Results

Example 7 : This example is taken from Wingo (1985).

$$\min_{y_1} \ y_1^6 - \frac{52}{25}y_1^5 + \frac{39}{80}y_1^4 + \frac{71}{10}y_1^3 - \frac{79}{20}y_1^2 - y_1 + \frac{1}{10}$$

$$-2 \leq y_1 \leq 11$$

This function has a local minimum at 0, with a value of $\frac{1}{10}$. The best solution reported by Wingo (1985) is -23627.1758, occurring at $y_1 = 11$. However, the global minimum of the function occurs at $y_1 = 10$, with an objective value of -29763.233 .

The problem can be converted to the desired form (15) with

$$s = 6, \quad a = (\frac{1}{10}, -1, -\frac{79}{20}, \frac{71}{10}, \frac{39}{80}, -\frac{52}{25}, 1), \quad and \;\; j = \emptyset \; .$$

The *qualifying* constraints for all the x variables are still of the form

$$y_1^k \geq 0, \quad or \;\; y_1^k < 0 \; ,$$

depending on whether, for a particular combination of bounds, the particular x_i variable is at its lower or upper bound. Hence, two **RD** problems are solved at every iteration. The bounds for x_i for these problems are given below :

(i) For the **RD** problem with $y_1^k \geq y_1$:

$$x_i^B = \begin{cases} x_i^L, & if \;\; \lambda_{i+1}^K \geq 0 \\ x_i^U, & if \;\; \lambda_{i+1}^K \leq 0 \end{cases} \qquad i = 1, 2, 3, 4 \; .$$

(ii) For the **RD** problem with $y_1^k < y_1$:

$$x_i^B = \begin{cases} x_i^U, & if \;\; \lambda_{i+1}^K \geq 0 \\ x_i^L, & if \;\; \lambda_{i+1}^K \leq 0 \end{cases} \qquad i = 1, 2, 3, 4 \; .$$

$$where \quad x^L = (-2, 0, -8, 0, -32), \quad x^U = (11, 121, 1331, 14641, 161051)$$

The **(GOP)** was applied from several different starting points. In every case, the algorithm converged to the global solution of -29763.233, taking around 175 iterations.

Example 8 :

This example is taken from Soland (1971).

$$\min_y \;\; -12y_1 - 7y_2 + y_2^2$$

$$
\begin{aligned}
subject\ to \quad -2y_1^4 + 2 - y_2 &= 0 \\
0 \le y_1 &\le 2 \\
0 \le y_2 &\le 3
\end{aligned}
$$

The nonconvexity in this problem comes from the presence of the polynomial term $-2y_1^4$ in the first constraint.

If the following transformation variables are introduced :

$$
\begin{aligned}
x_1 &= y_1 \\
x_2 &= x_1 y_1 = y_1^2 \\
x_3 &= x_2 y_1 = y_1^3
\end{aligned}
$$

Then, by fixing $y_1 = y_1^k$, the primal problem at the kth iteration becomes

$$
\min_{x,y_2} \; -12y_1^k - 7y_2 + y_2^2
$$

$$
\begin{aligned}
subject\ to \quad x_1 - y_1^k &= 0 \\
x_2 - x_1 y_1^k &= 0 \\
x_3 - x_2 y_1^k &= 0 \\
-2y_1^k x_3 + 2 - y_2 &= 0 \\
0 - y_2 &\le 0 \\
y_2 - 3 &\le 0
\end{aligned}
$$

The solution of this problem gives the Lagrange multipliers λ_a^k, λ_b^k and λ_c^k corresponding to the three new equality constraints introduced, λ_d^k for the original equality constraint, and μ_l^k and μ_u^k for the lower and upper bound constraints for y_2.

The Lagrange function formulated from this problem can be written as

$$
\begin{aligned}
L(x, y, \lambda^k, \mu^k) &= -12y_1 - 7y_2 + y_2^2 + \lambda_a^k(x_1 - y_1) + \lambda_b^k(x_2 - x_1 y_1) \\
&\quad + \lambda_c^k(x_3 - x_2 y_1) + \lambda_d^k(-2x_3 y_1 + 2 - y_2) + \mu_l^k(0 - y_2) + \mu_u^k(y_2 - 3)
\end{aligned}
$$

The KKT gradient conditions for the kth primal problem are

$$
\begin{aligned}
\nabla_{x_1} L(x, y_2, y_1^k, \lambda^k, \mu^k) &= \lambda_a^k - \lambda_b^k y_1^k = 0 \\
\nabla_{x_2} L(x, y_2, y_1^k, \lambda^k, \mu^k) &= \lambda_b^k - \lambda_c^k y_1^k = 0 \\
\nabla_{x_3} L(x, y_2, y_1^k, \lambda^k, \mu^k) &= \lambda_c^k - 2\lambda_d^k y_1^k = 0 \\
\nabla_{y_2} L(x, y_2, y_1^k, \lambda^k, \mu^k) &= 2y_2^k - 7 - \lambda_d^k - \mu_l^k + \mu_u^k = 0
\end{aligned}
$$

Using these conditions, the Lagrange function (after linearizing the terms in y_2 around y_2^k) can be reformulated as

$$\begin{aligned} L(x,y,\lambda^k,\mu^k)|_{y_2^k}^{lin} \;=\; & \lambda_b^k(y_1^k - y_1)x_1 + \lambda_c^k(y_1^k - y_1)x_2 + 2\lambda_d^k(y_1^k - y_1)x_3 \\ & -{y_2^k}^2 - (12+\lambda_a^k)y_1 + 2\lambda_d^k - 3\mu_u^k. \end{aligned}$$

It can be seen that the *connected* variables are x_1, x_2, and x_3. The *qualifying* constraints for x_1, x_2, and x_3 are all of the same form except for their sign. This means that only two *relaxed dual* problems need to be solved at every iteration. The bounds for x for these problems are shown below :

$$For \quad y_1^K - y_1 \geq 0 \;:$$

$$x_1^B = \begin{cases} x_1^L, & if \;\; \lambda_2^K \geq 0 \\ x_1^U, & if \;\; \lambda_2^K \leq 0 \end{cases}$$

$$x_2^B = \begin{cases} x_2^L, & if \;\; \lambda_3^K \geq 0 \\ x_2^U, & if \;\; \lambda_3^K \leq 0 \end{cases}$$

$$x_3^B = \begin{cases} x_3^L, & if \;\; \lambda_4^K \geq 0 \\ x_3^U, & if \;\; \lambda_4^K \leq 0 \end{cases}$$

$$For \quad y_1^K - y_1 < 0 \;:$$

$$x_1^B = \begin{cases} x_1^U, & if \;\; \lambda_2^K \geq 0 \\ x_1^L, & if \;\; \lambda_2^K \leq 0 \end{cases}$$

$$x_2^B = \begin{cases} x_2^U, & if \;\; \lambda_3^K \geq 0 \\ x_2^L, & if \;\; \lambda_3^K \leq 0 \end{cases}$$

$$x_3^B = \begin{cases} x_3^U, & if \;\; \lambda_4^K \geq 0 \\ x_3^L, & if \;\; \lambda_4^K \leq 0 \end{cases}$$

The **GOP** algorithm was applied to the problem in this form. From a starting point of 0 for y_1, the algorithm converged to the global solution of -16.73889 at $y = (0.7175, 1.47)$ in 89 iterations, solving 3 subproblems at every iteration.

Example 9:

This is a test example constructed to illustrate the application of the **GOP** algorithm when the feasible region consists of two disconnected sub-regions.

Consider the following problem :

$$\begin{aligned} \min_{y} \quad & -y_1 - y_2 \\ y_2 \;\leq\; & 2 + 2y_1^4 - 8y_1^3 + 8y_1^2 \\ y_2 \;\leq\; & 4y_1^4 - 32y_1^3 + 88y_1^2 - 96y_1 + 36 \\ 0 \;\leq\; & y_1 \leq 3 \\ 0 \;\leq\; & y_2 \leq 4 \end{aligned}$$

The constraint region for this problem is given in Figure 3(a). As can be seen, there are two distinct regions where the problem is feasible. Because of this reason, if a conventional **NLP** solver were applied to this problem, it is highly unlikely that the solver would converge to the global solution at point C. Depending on the starting point, the solution will be one of the points A, B, or C.

Consider the application of the **GOP** algorithm to this example. The following *transformation* variables are defined first :

$$\begin{aligned} x_1 &= y_1 \\ x_2 &= y_1^2 = x_1 y_1 \\ x_3 &= y_1^3 = x_2 y_1 \end{aligned}$$

Then, by *projecting* on y_1, the following linear primal problem is obtained at the kth iteration :

$$\min_{x_1, y_2} -x_1 - y_2$$

$$\begin{aligned} -2y_1^k x_3 + 8x_3 - 8x_2 + y_2 - 2 &\leq 0 \\ -4y_1^k x_3 + 32x_3 - 88x_2 + 96x_1 + y_2 - 36 &\leq 0 \\ x_1 - y_1^k &= 0 \\ x_2 - y_1^k x_1 &= 0 \\ x_3 - y_1^k x_2 &= 0 \end{aligned}$$

The optimal solution of the primal problem as a function of y_1 is plotted in Figure 3(b).

From the solution of the primal problem for a fixed value of $y = y^k$, and through the use of the KKT conditions for the primal problem, the Lagrange function can be formulated as

$$\begin{aligned} L(x, y, \lambda^k, \mu^k) = & \ \lambda_2^k (y_1^k - y_1) x_1 + \lambda_3^k (y_1^k - y_1) x_2 + (2\mu_1^k + 4\mu_2^k)(y_1^k - y_1) x_3 \\ & -2\mu_1^k - 36\mu_2^k - \lambda_1^k y_1 \end{aligned}$$

where λ^k are the multipliers for the equality constraints due to the introduction of the x variables, and μ_1^k and μ_2^k are the multipliers corresponding to the original inequality constraints.

Thus, the *qualifying* constraints to be added to the relaxed dual **RD** problem are of the form $y_1^k - y_1$ being greater than or less than zero, with the actual form being determined by the nature of the bound for the corresponding x variable. Hence, there are two relaxed dual problem solved at every iteration. These two problems provide underestimators for the optimal solution for values of y both less than and greater than y^k. Hence, this ensures that the difficulty of crossing from one part of the feasible region to the other part is removed. Also, it is ensured that for every value of y, there will be a corresponding **RD** problem (solved from some y^k near that y which provides a stored solution *as well as* an underestimating Lagrange function.

The global solution of -5.50796 (occurring at $y_1 = 2.3295$, point C on Figure 3(a)) was determined from all starting points on applying the **GOP** algorithm. From a starting point of 0 for y_1, the algorithm took 210 iterations to converge.

6 Conclusions

The **GOP** algorithm represents a generalized approach for determining a global optimum for different classes of nonconvex programming problems. The application of the algorithm to quadratic programming problems with linear and/or quadratic constraints, and to problems involving polynomial objective function and/or constraints has been presented in this paper. For each of the classes of problems discussed in the paper, it is possible to formulate a very general algorithm for solving problems of that class, with specific details for a problem being required only in terms of the data for the problem. Indeed, it is possible to develop a general mathematical model that can be applied to problems involving any combination of quadratic and polynomial terms in the objective function and/or constraints.

It should be noted that the overriding factor in the computational time requirement lies in the solution of the relaxed dual problem. Since this is directly related to the number of *connected* variables, the **GOP** algorithm is computationally efficient when the number of *connected* variables is small compared to the total number of variables, as shown by the pooling and blending problems.

An important consideration in the proposed **GOP** algorithm is that the relaxed dual subproblems solved at every iteration are structurally identical, except for a change in some parameters, namely the bounds used for the x variables and the sign of the *qualifying* constraints. Since the solution of these problems are independent of each other, the algorithm is ideally suited for use of parallel processing. This can be of great significance for problems involving a large number of variables.

Research work on (a) the use of parallel computing for the **GOP** algorithm and (b) the development of additional theoretical properties that can enhance its computational efficiency are currently underway, and results will be reported in a future publication.

Acknowledgement

The authors gratefully acknowledge financial support from the National Science Foundation under Grant CBT-8857013, as well as support from Amoco Chemical Co., Tennessee Eastman Co., and Shell Development Co.

References

[1] Dixon, L.C.W., and Szego, G.P., Editors, *Towards Global Optimization*, North Holland, Amsterdam (1975).

[2] Floudas, C.A., and Aggarwal, A., A Decomposition Strategy for Global Optimum Search in the Pooling Problem, *Operations Research Journal on Computing*, **2(3)**, in Press (1990).

[3] Floudas, C.A., Aggarwal, A., and Ciric, A.R., Global Optimum Search for Nonconvex NLP and MINLP problems, *Computers and Chemical Engineering*, **13**, 1117 (1989).

[4] Floudas, C.A., and Pardalos, P.M., A Collection of Test Problems for Constrained Global Optimization Algorithms, *Lecture Notes in Computer Science*, Springer-Verlag, in Press (1990).

[5] Floudas, C.A., and Visweswaran, V., A Global Optimization Algorithm **(GOP)** for Certain Classes of Nonconvex NLPs : I. Theory, *Submitted For Publication.*

[6] Hansen, P., Jaumard, B., and Lu, S-H., An Analytical Approach to Global Optimization, *To appear in Mathematical Programming* (1990).

[7] Haverly, C.A., Studies of the Behaviour of Recursion for the Pooling Problem, *SIGMAP Bulletin*, **25**, 19 (1978).

[8] Haverly, C.A., Behaviour of Recursion Model - More Studies, *SIGMAP Bulletin*, **26**, 22 (1979).

[9] Hesse, R., A Heuristic Search Procedure for Estimating a Global Solution of Nonconvex Programming Problems, *Operations Research*, **21**, 1267 (1973).

[10] Konno, H., A Cutting Plane Algorithm for Solving Bilinear Programs, *Mathematical Programming*, **11**, 14 (1976).

[11] Lasdon, L.S., Waren, A.D., Sarkar, S., and Palacios, F., Solving the Pooling Problem Using Generalized Reduced Gradient and Successive Linear Programming Algorithms, *ACM SIGMAP Bulletin*, **27**, 9 (1979).

[12] Soland, R.M., An Algorithm For Separable Nonconvex Programming Problems II : Nonconvex Constraints, *Management Science*, **17**, (1971).

[13] Wingo, D.R., Globally Minimizing Polynomials Without Evaluating Derivatives, *International Journal of Computer Mathematics*, **17**, 287 (1985).

[14] Zwart, P.B., Nonlinear Programming : Counterexamples to Two Global Optimization Algorithms, *Operations Research*, **21**, 1260 (1973).