An International Journal of Optimization
and Control: Theories & Applications
Vol.4, No.2, pp.89-98 (2014) © IJOCTA
ISSN:2146-0957 eISSN:2146-5703
DOI:10.11121/ijocta.01.2014.00171
http://www.ijocta.com

Approximate solution algorithm for multi-parametric non-convex programming problems with polyhedral constraints

Abay Molla Kassa[a] and Semu Mitiku Kassa[b]

[a]Department of Chemical Engineering, Addis Ababa Institute of Technology
P.O.Box 1176, Addis Ababa, Ethiopia
Email: abay.molla@aait.edu.et

[b]Department of Mathematics, Addis Ababa University
P.O.Box 1176, Addis Ababa, Ethiopia
Email: semu.mitiku@aau.edu.et

(Received January 16, 2013; in final form June 12, 2014)

Abstract. In this paper, we developed a novel algorithmic approach for the solution of multi-parametric non-convex programming problems with continuous decision variables. The basic idea of the proposed approach is based on successive convex relaxation of each non-convex terms and sensitivity analysis theory. The proposed algorithm is implemented using MATLAB software package and numerical examples are presented to illustrate the effectiveness and applicability of the proposed method on multi-parametric non-convex programming problems with polyhedral constraints.

Keywords: Multi-parametric Programming; Convex relaxation.
AMS Classification: 90B50, 90C26, 90C31, 90C99.

1. Introduction

Variability and uncertainty emerges in all different levels of industry from the detailed process description to multi-site manufacturing. The existence of parameter variability in most real process design and operation problems necessitates the consideration of uncertainties in the mathematical programming models used to simulate and optimize the design, performance and process operations [1]. Many process engineering problems involve varying parameters such as attributed to fluctuations in resource, market requirements, prices and so on, which can affect the feasibility and economics of the project [2], [3].

According to the parameters description, different solution approaches have been proposed. Among these parametric programming approach is the one which is based on the sensitivity analysis theory. Sensitivity analysis provides solutions in the neighborhood of the nominal value of the varying parameters, whereas parametric programming provides a complete map of the optimal solution in the space of the varying parameters [3].

The key advantage of multi-parametric programming approach is that the optimal solution is obtained as a function of the varying parameters without exhaustively enumerating the entire space of the varying parameters [2].

Algorithms and applications of multi-parametric programming approaches have been extensively studied in the literature [1, 2, 4, 5, 6, 7] but all existing algorithms are limited to convex multi-parametric programming problems even if practical problems may include non-convex formulation. In the last decade, Pistikopous *et al.*, [3] proposed algorithmic strategy

Corresponding Author. Email: semu.mitiku@aau.edu.et.

for parametric non-convex programming problems, but in practice the proposed algorithm is costly and limited to bilinear terms in addition to linear terms in the objective function as well as in constraint sets and a single perturbation parameter. It is difficult to use the proposed algorithm, even when there exist convex nonlinear terms in the objective in addition to bilinear terms, because of the difficulty in the algorithm to compare nonlinear functions of parameters. Furthermore, the efficiency of the algorithm in [3] is highly affected when the parameters are vectors instead of single parameter. In general, existing algorithmic conditions do not hold for general multi-parametric non-convex programming problems.

In this paper we have developed a novel global optimization technique, by modifying the key procedures in [3], for solving a more general multi-parametric non-convex programming problems using sensitivity analysis theory which overcomes the limitations of the existing algorithmic methods. In particular, the solution approach effectively solves parametric problems with any twice continuously differentiable nonlinear objective function and having polyhedral constraints.

This paper is organized as follows: in section 2, some preliminary concepts such as: convexification of non-convex terms and the theory of non-linear multi-parametric programming problem with respective critical regions are described. Moreover, the major difficulties associated with obtaining global parametric solution using the existing methods have been discussed in this section. The proposed algorithm for the solution of multi-parametric non-convex programming problems with affine constraints is then described in Section 3, and illustrative examples are presented. The paper ends with conclusive remarks in Section 4.

2. Preliminary

2.1. Convex relaxation

If the optimization problem contains non-convex terms, the Karush-Kuhn-Tucker (KKT) conditions may not produce optimal solutions to such problems. To apply the KKT conditions as solution mechanism to non-convex problems, the occurrence of any non-convex term both in the objective as well as in the constraint must be underestimated by a convex envelope to approximate it by a convex function (see, for instance, [8, 9, 10]).

The convex envelope of bilinear terms $b_{ij}x_ix_j$ taken over the rectangle $R = \{(x_i, x_j) : x_i^L \leq x_i \leq x_i^U, x_j^L \leq x_j \leq x_j^U\}$ is denoted by $VexR[b_{ij}x_ix_j]$ and can be found as follows:

Theorem 1. [8] *Let b_{ij}, for $i = 1, 2, \dots, n-1$ and $j = i+1, \dots, n$, be a real number, then the convex envelope of a bilinear term $b_{ij}y_iy_j$ can be formulated as:*

$$VexR[b_{ij}x_ix_j] = max\{b_{ij}l_{ij}^1(x_i, x_j), b_{ij}l_{ij}^2(x_i, x_j)\}$$

where,

$$l_{ij}^1(x_i, x_j) = \begin{cases} x_j^L x_i + x_i^L x_j - x_i^L x_j^L, b_{ij} > 0 \\ x_j^U x_i + x_i^L x_j - x_i^L x_j^U, b_{ij} \leq 0 \end{cases}$$

and

$$l_{ij}^2(y_i, y_j) = \begin{cases} x_j^U x_i + x_i^U x_j - x_i^U x_j^U, b_{ij} > 0 \\ x_j^L x_i + x_i^U x_j - x_i^U x_j^L, b_{ij} \leq 0 \end{cases}$$

Univariate concave functions can be trivially underestimated by their linearization at the lower bound of the variable range [9]. Thus, the convex envelope of the concave function $f(x)$ over the interval $[x^L, x^U]$ is the linear function of x:

$$CEnve = f(x^L) + \frac{f(x^U) - f(x^L)}{x^U - x^L}(x - x^L) \quad (1)$$

All other general non-convex terms for which customized lower bound do not exist are underestimated as proposed in [9]. A generic non-convex function $f(x)$ is underestimated over the entire domain $[x^L, x^U]$ by the function $F(x)$ and defined as:

$$F(x) = f(x) + \alpha \sum_{i=1}^{n} (x_i^L - x_i)(x_i^U - x_i) \quad (2)$$

where α is a positive scalar and is given by: $\alpha \geq \max\{0, \min_{x^L \leq x \leq x^U} \lambda_i(x)\}$, where the λ_i's are eigenvalues of the Hessian matrix $(H_f(x))$ of $f(x)$.

Theorem 2 (Properties of $F(x)$)**.**

(i) *$F(x)$ is a valid under-estimator of $f(x)$*
(ii) *$F(x)$ matches $f(x)$ at all corner points.*
(iii) *$F(x)$ is convex in $x_i \in [x^L, x^U]$, $i = 1, 2, \dots, n$.*
(iv) *The maximum separation between the non-convex term of generic structure $f(x)$ and its convex relaxation $F(x)$ is bounded and proportional to the positive parameters α_i and to the square of the diagonal of the current box constraints*

Proof: see [10]

2.2. Multi-parametric nonlinear programming problem

Consider the general parametric nonlinear programming problem:

$$
\begin{aligned}
Z(\theta) &= \min_x f(x,\theta) \\
&\text{s.t.} \\
&g_i(x,\theta) \le 0, \text{ for all } i = 1,2,\dots,p, \\
&h_j(x,\theta) = 0, \text{ for all } j = 1,2,\dots,q \\
&\qquad x \in X \subseteq \mathbb{R}^n, \theta \in \Theta \subseteq \mathbb{R}^m,
\end{aligned} \tag{3}
$$

where f, g's and h's are twice continuously differentiable in x and θ. Assume also that f is a convex function and g_i's, h_j's define a convex set.

The first-order KKT optimality conditions for (3) are given as follows:

$$
\begin{aligned}
L &= f(x,\theta) + \sum_{i=1}^{p} \lambda_i g_i(x,\theta) \\
&\quad + \sum_{j=1}^{q} \mu_j \; h_j(x,\theta), \\
\nabla_x L &= 0, \\
g_i(x,\theta) &\le 0, \lambda_i g_i(x,\theta) = 0, \\
\lambda_i &\ge 0, \text{ for all, } i = 1,2,\dots,p \\
h_j(x,\theta) &= 0, \text{ for all, } j = 1,2,\dots,q
\end{aligned} \tag{4}
$$

where, λ and μ are vectors of Lagrange multipliers. The main sensitivity result for (3) is derived directly from system (4) and is given in the next theorem.

Theorem 3. [11] *Let θ_0 be a vector of parameter values and (x_0, λ_0, μ_0) be a KKT triple corresponding to (4), where, λ_0 is nonnegative and x_0 is feasible in (3). Also assume that,*

(1) *Strict complementary slackness (SCS) condition holds,*
(2) *The gradients of all binding constraints are linearly independent (LICQ: Linear Independence Constraint Qualification condition holds), and*
(3) *The second-order sufficiency conditions (SOSC) hold.*

Then, in the neighborhood of θ_0, there exists a unique, once continuously differentiable function, $Z(\theta) = [x(\theta), \lambda(\theta), \mu(\theta)]$, satisfying (4) with $Z(\theta_0) = [x(\theta_0), \lambda(\theta_0), \mu(\theta_0)]$, where $x(\theta)$ is a unique isolated minimizer for (3), and

$$
\begin{pmatrix} \frac{dx(\theta_0)}{d\theta} \\ \frac{d\lambda(\theta_0)}{d\theta} \\ \frac{d\mu(\theta_0)}{d\theta} \end{pmatrix} = -M_0^{-1} \cdot N_0 \tag{5}
$$

where, M_0 and N_0 are the Jacobian of system (4) with respect to x and θ:

$$
M_0 = \begin{bmatrix}
\nabla_{xx} L & \nabla_x g_1 \cdots & & \nabla_x g_p \\
\lambda_1 \nabla_x^T g_1 & -g_1 & & 0 \\
\vdots & & \ddots & \\
\lambda_p \nabla_x^T g_p & & -g_p & 0 \\
\nabla_x^T h_1 & 0 \cdots & & 0 \\
\vdots & & & \\
\nabla_x^T h_q & 0 \cdots & & 0
\end{bmatrix}
$$

$$
\begin{aligned}
N_0 &= \big(\nabla_{\theta x}^2 L, -\lambda_1 \nabla_\theta g_1, \dots, -\lambda_p \nabla_\theta g_p, \\
&\qquad -\nabla_\theta h_1, \dots, -\nabla_\theta h_q\big)^T
\end{aligned}
$$

Note that the assumptions in Theorem 3 ensure that the inverse of the Jaccobian of Equation (5) exists [4, 6, 3]. In other words, when M_0 is not invertible any violation of assumptions in Theorem 3 is easily detected.

In [6] Dua *et al.*, have proposed an algorithm to solve Equ. (5) in the entire range of the varying parameters for general convex problems. This algorithm is based on approximations of the nonlinear optimal expression, $x = \gamma^\star(\theta)$ by a set of first order approximations.

Corollary 1. *(First order estimation of $x(\theta)$, $\lambda(\theta)$, $\mu(\theta)$, near $\theta = \theta_0$ [12]). Under the consideration of Theorem 3, a first order approximation of $[x(\theta), \lambda(\theta), \mu(\theta)]$ in the neighborhood of θ_0 is,*

$$
\begin{bmatrix} x(\theta) \\ \lambda(\theta) \\ \mu(\theta) \end{bmatrix} = \begin{bmatrix} x_0 \\ \lambda_0 \\ \mu_0 \end{bmatrix} - M_0^{-1} \cdot N_0 \cdot (\theta - \theta_0) \tag{6}
$$

where $(x_0, \lambda_0, \mu_0) = (x(\theta_0), \lambda(\theta_0), \mu(\theta_0)), M_0 = M(\theta_0), N_0 = N(\theta_0)$

The space of θ where this solution (6) remains optimal is defined as the *critical region*, CR, and can be obtained by using feasibility and optimality conditions. Feasibility is ensured by substituting $x(\theta)$ into the inactive inequalities given in problem (3), whereas the optimality condition is given by $\check{\lambda}(\theta) \ge 0$, where $\check{\lambda}(\theta)$ corresponds to the vector of active inequalities, resulting in a set of parametric constraints. Each piecewise linear approximation is confined to regions defined by feasibility and optimality conditions [6]. If $\check{g}$ corresponds to the non-active constraints, and $\check{\lambda}$ to the lagrangian multipliers of the active constraints:

$$
CR = \begin{cases} \check{g}(x(\theta), \theta) \le 0 & \text{Feasibility conditions} \\ \check{\lambda}(\theta) \ge 0 & \text{Optimality conditions.} \end{cases}
$$

Consequently, the explicit expressions are given by a conditional piecewise linear function

[6]:

$$\begin{cases} x = C_1 + K_1 \cdot \theta, & \text{if } \theta \in CR^1 \\ x = C_2 + K_2 \cdot \theta, & \text{if } \theta \in CR^2 \\ \vdots \quad\quad \vdots & \quad \vdots \\ x = C_p + K_p \cdot \theta, & \text{if } \theta \in CR^p \end{cases}, \tag{7}$$

where C_i are column vectors and K_i are real matrices, whereas $CR^i \in \mathbb{R}^m$ are critical regions and note that CR^i denotes the i^{th} critical region.

For problems involving convex f, g and h, the parametric solutions, as described in equation (7) within the corresponding critical regions, are necessary and sufficient. As a result, the existing algorithms which are proposed in [1, 2, 4, 5, 6, 7] work efficiently. However, for the general nonlinear case, the convexity assumption of f, g and h are usually extended to include non-convex cases as long as KKT necessary conditions are met and hence, the KKT conditions can no longer guarantee global optimality of the problem for fixed parameter $\theta = \theta_0$. Hence, methods based on sensitivity theory for the solution of general nonlinear multi-parametric programming problems bound to local.

Recently, Pistikopoulos *et al.*, [3] have proposed a solution strategy for special non-convex multi-parametric programming problems based on a branch-and-bound algorithm to locate the global parametric solution. It may work well when the objective function contains only affine functions with respect to x and θ explicitly, where θ is a scalar parameter. When non-linear (even polynomial) functions appear at the objective function the bounding procedure becomes unmanageable. The complexity further increases significantly, as the number of parameters increase. A further difficulty arises when there exists general non-convex terms in addition to special non-convex terms at the objective function defined in polyhedral region.

In the following section we have proposed a new algorithmic approach which can be applied to approximate the solution of multi-parametric non-convex programming problems.

3. The Proposed Approach

3.1. Mathematical aspects of the proposed algorithm

In this section we have presented a solution approach for non-convex multi-parametric programming problems which is optimized over a convex polyhedral region. But, the algorithm can be extended to any non-linear constraint functions by taking each non-linear constraints to the objective function using penalty terms.

To proceed the presentation, consider the following multi-parametric non-convex programming problem:

$$\begin{aligned} Z(\theta) &= \min_x \left\{ f_n(x,\theta) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} b_{ij} x_i x_j \right. \\ &\qquad\qquad \left. + cf(x,\theta) + c(x,\theta) \right\} \\ \text{s.t.} \quad & g(x,\theta) \le 0 \\ & h(x,\theta) = 0, \\ & x^L \le x \le x^U, \\ & \theta^L \le \theta \le \theta^U \end{aligned} \tag{8}$$

where, f_n and c are generic non-convex and linear combination of concave functions in x, whereas cf is a convex term and g and h define a convex polyhedron.

Thus, problem (8) can be reduced to a standard non-linear programming problem by fixing a feasible value to a parameter vector, $\theta = \theta_0$;

$$\begin{aligned} Z(\theta_0) &= \min_x \left\{ f_n(x,\theta_0) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} b_{ij} x_i x_j \right. \\ &\qquad\qquad \left. + cf(x,\theta_0) + c(x,\theta_0) \right\} \\ \text{s.t.} \quad & g(x,\theta_0) \le 0 \\ & h(x,\theta_0) = 0, \\ & x^L \le x \le x^U \end{aligned} \tag{9}$$

As discussed in Subsection 2.1, each generic non-convex, bilinear, and concave terms can be under-estimated by $F(x,\theta_0)$, $VexR[b_{ij}x_ix_j]$ and $CEnve$ respectively. Thus, the convex under-estimator of problem (9) can be formulated as:

$$\begin{aligned} Z(\theta_0) &= \min_x \left\{ F(x,\theta_0) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} VexR[b_{ij} x_i x_j] \right. \\ &\qquad\qquad \left. + cf(x,\theta_0) + CEnve \right\} \\ \text{s.t.} \quad & g(x,\theta_0) \le 0 \\ & h(x,\theta_0) = 0, \\ & x^L \le x \le x^U \end{aligned} \tag{10}$$

Then one can solve problem (9) locally to obtain an upper bound, $\hat{Z}$, and problem (10) to obtain a lower bound, $\check{Z}$, of the exact solution of the original problem around θ_0. If problem (10) is infeasible and its objective value is above $\hat{Z}$, we fathom the region for θ_0. Otherwise, one compares $\hat{Z}$ and $\check{Z}$ (*note that both of them are real numbers*). If the difference is greater than the pre-defined tolerance error, ϵ, the lower bound, $\check{Z}$ will be stored along with the solution set and the optimization variable, with the longest side from among those which contribute to the non-convexity of the problem, will be branched as follows:

$$R_1 = \begin{bmatrix} x_1^L & x_1^U \\ x_2^L & x_2^U \\ \vdots & \vdots \\ x_i^L & \frac{(x_i^L + x_i^U)}{2} \\ \vdots & \vdots \\ x_n^L & x_n^U \end{bmatrix}$$

and,

$$R_2 = \begin{bmatrix} x_1^L & x_1^U \\ x_2^L & x_2^U \\ \vdots & \vdots \\ \frac{(x_i^L + x_i^U)}{2} & x_i^U \\ \vdots & \vdots \\ x_n^L & x_n^U \end{bmatrix}$$

Solving problem (9) over both sub-rectangles locally gives the upper bounds. By comparing the two upper bounds with the previous upper bound the minimum of them will be taken as a current upper bound $C\hat{Z}$.

Again problem (9) will be under-estimated inside each of the two sub-rectangles R_1 and R_2 as:

$$\begin{aligned} Z(\theta_0) &= \min_x \left\{ F(x,\theta_0) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} VexR[b_{ij}x_i x_j] \right. \\ &\qquad \left. + cf(x,\theta_0) + CEnve \right\} \\ \text{s.t.} \quad & g(x,\theta_0) \le 0 \\ & h(x,\theta_0) = 0, \\ & x \in R_1 \end{aligned} \tag{11}$$

and,

$$\begin{aligned} Z(\theta_0) &= \min_x \left\{ F(x,\theta_0) + \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} VexR[b_{ij}x_i x_j] \right. \\ &\qquad \left. + cf(x,\theta_0) + CEnve \right\} \\ \text{s.t.} \quad & g(x,\theta_0) \le 0 \\ & h(x,\theta_0) = 0, \\ & x \in R_2 \end{aligned} \tag{12}$$

Then one solves the resulting problems and stores the objective values along with the solution set, if the obtained solution is feasible and objective values are less than the upper bound $\hat{Z}$. Otherwise the respective sub-rectangle for $\theta = \theta_0$ will be fathomed. The minimum over the stored solution set, $\check{Z}$, will be chosen as the lower bound and it will be compared with the current upper bound $\hat{Z}$. If $\hat{Z} - \check{Z} > \epsilon$, the lower bound will be discarded and the same procedure will be repeated as discussed above with working variable bound corresponds to the previously found minimum lower bound, (i.e., the variable box corresponding to the discarded minimum value should be further branched to obtain another lower bounds) until the difference falls below the pre-defined tolerance error, (i.e., $\hat{Z} - \check{Z} \le \epsilon$).

Theorem 4. *The difference between $\hat{Z}$ and $\check{Z}$ converges within finite number of branching steps.*

Proof. To prove the statement of the theorem we need to show that the lower bound $\check{Z}$ is an increasing sequence. To this end, let $\check{Z}_k$ denote the lower bound for k^{th} iteration (k^{th} branching step). As a result of the above procedure, we have the current lower bound defined by: $C\check{Z}_k = \min\{\min\{\check{Z}_i\}_{i=1}^{k-1}, \check{Z}_k\}$ and the selected minimum value has to be discarded if it does not satisfy the stopping criteria. Hence, we have large values in the solution set. Similarly, at $k+1$ we have, $C\check{Z}_{k+1} = \min\{\min\{\check{Z}_i\}_{i=1}^{k} \setminus C\check{Z}_k, \check{Z}_{k+1}\}$, (since $C\check{Z}_k$ has already been discarded from the set) and as the size of the rectangular domain (like R_1 and R_2) decreases, the maximum separation between the original non-convex function and its respective convex under-estimator function decreases. This implies that $C\check{Z}_{k+1} \ge C\check{Z}_k$. This shows that the lower bound is increasing.

On the other hand, let $\hat{Z}_k$ denote the upper bound for k^{th} iteration. Again from the above mathematical procedure, we have the current upper bound, $C\hat{Z}_k = \min\{\min\{\hat{Z}_i\}_{i=1}^{k-1}, \hat{Z}_k\}$ and at the $(k+1)^{th}$ step we have the current upper bound, $C\hat{Z}_{k+1} = \min\{\min\{\check{Z}_i\}_{i=1}^{k}, \hat{Z}_{k+1}\}$, but, $C\hat{Z}_k = \min\{\hat{Z}_i\}_{i=1}^{k}$. This implies, $C\hat{Z}_{k+1} = \min\{C\hat{Z}_k, \hat{Z}_{k+1}\}$. Hence, we have a decreasing sequence of upper bounds, i.e., $C\hat{Z}_k \ge \hat{Z}_{k+1}$

Therefore, since $C\hat{Z}_k(x)$ decreases as k increases and $C\check{Z}_k(x)$ increases as k increases, we can conclude that the difference is a decreasing sequence and hence it converges within finite step. □

If the difference falls below the given tolerance error at the k^{th} branching step (sub-rectangle), the approximate solution of the k^{th} under-estimator subproblem is taken as a KKT triple (x_0, λ_0, μ_0) for $\theta = \theta_0$ and the problem itself to define approximate parametric solution to the original problem in the neighborhood of θ_0

as;

$$\begin{bmatrix} x(\theta) \\ \lambda(\theta) \\ \mu(\theta) \end{bmatrix} = \begin{bmatrix} x_0 \\ \lambda_0 \\ \mu_0 \end{bmatrix} - M_0^{-1} \cdot N_0 \cdot (\theta - \theta_0), \quad \theta \in CR \tag{13}$$

where, CR is a convex polyhedron which is called *a critical region* and can be obtained as described in Subsection 2.2

Theorem 5. *Expression* (13) *is an approximate parametric solution of problem* (8) *over the corresponding critical region, CR.*

Proof. From the discussion above the KKT triple (x_0, λ_0, μ_0) is an approximate solution for problem (8) for $\theta = \theta_0$. Hence by recalling Theorem 3 and Corollary 1, we have that expression (13) is an approximate parametric solution of problem (8). □

If CR has not covered the parametric region, we repeat again the same mathematical procedure as in above with any new feasible parameter $(\theta = \theta_0)$ taken from the rest of parametric regions until the parametric region has been explored successfully.

To define the rest of the parametric region, consider $CR_{IG} = [\theta^L, \theta^U]$ to be the overall parametric region and let the inequalities $\{c_1 \leq 0, c_2 \leq 0, c_3 \leq 0\}$ define CR. Now the rest of the parametric region can be defined as $CR^{\text{rest}} = CR_{IG} \setminus CR$, which can be obtained by reversing the inequalities in CR one-by-one. For example, consider inequality $c_1 \leq 0$, the rest of the region can be addressed by reversing the sign of inequality $c_1 \leq 0$ and removing redundant constraints in CR_{IG}, which is $CR_1^{\text{rest}} = \{c_1 \geq 0, \theta_1 \geq \theta_2^L, \theta_2 \leq \theta_2^U\}$ where, $\theta = (\theta_1, \theta_2)$. Thus by considering the rest of the inequalities, the total of the rest region is given by, $CR^{\text{rest}} = \{CR_1^{\text{rest}} \cup CR_2^{\text{rest}} \cup CR_3^{\text{rest}}\}$, where CR_1^{rest}, CR_2^{rest} and CR_3^{rest} are given in Table 1.

Table 1. Definition of the rest regions

Region	Inequalities	
CR_1^{rest}	$c_1 \geq 0,$	$\theta_1 \geq \theta_1^L, \quad \theta_2 \leq \theta_2^U$
CR_2^{rest}	$c_1 \leq 0, \quad c_2 \geq 0,$	$\theta_1 \leq \theta_1^U, \quad \theta_2 \leq \theta_2^U$
CR_3^{rest}	$c_1 \leq 0, c_2 \leq 0, c_3 \geq 0,$	$\theta_1^L \leq \theta_1 \leq \theta_2^U, \theta_2^L \leq \theta_2$

Theorem 6. *Let $X \subseteq \mathbb{R}^m$ be a polyhedron and $CR^Q = \{x \in X : \tilde{g}_2(x) - \tilde{b} \leq 0\} \subseteq X$, be a critical region. Assume $CR^Q \neq \emptyset$. Also let $CR^i = \{x \in X : \tilde{g}_2^i(x) - \tilde{b}^i > 0, \tilde{g}_2^j(x) - \tilde{b}^j \leq 0, \forall j < i,\ i = 1, 2, \ldots, K\}$ where $K = size(b)$, and let $CR^{rest} = \bigcup_{i=1}^K CR^i$. Then*

(1) $CR^{rest} \bigcup CR^Q = X$,
(2) $CR^Q \bigcap CR^i = \emptyset$,
(3) $CR^i \bigcap CR^j = \emptyset, \ \forall i \neq j$, *i.e.* $\{CR^Q, CR^1, \ldots, CR^K\}$ *is a partition of* X.

Proof. (1) Since $CR^i \subseteq X$ for all i and $CR^Q \subseteq X$, it is clear that $CR^{rest} \bigcup CR^Q \subseteq X$. To show the backward inclusion let $x \in X$ and assume that $x \notin CR^Q$. Then, there exists an index i such that $\tilde{g}_2^i(x) - \tilde{b}^i > 0$. Let $i^* = \min_{i \leq K}\{i : \tilde{g}_2^i(x) > \tilde{b}^i\}$, by definition of i^* we have $\tilde{g}_2^{i^*}(x) > \tilde{b}^{i^*}$ and $\tilde{g}_2^j(x) < \tilde{b}^j$, $\forall j < i^*$. This implies that $x \in CR^{i^*}$, thus $x \in CR^{Rest} \bigcup CR^Q$. Hence $CR^{rest} \bigcup CR^Q = X$.

(2) If $x \in CR^Q$ then by definition, there doesn't exist an index i that satisfy $\tilde{g}_2^i(x) - \tilde{b}^i > 0$. which implies that $x \notin CR^i$.

(3) Let $x \in CR^i$ and take $i > j$. Since $x \in CR^i$, by definition of $CR^i (i > j)$ $\tilde{g}_2^j(x) - \tilde{b}^j \leq 0$, which implies that $x \notin CR^j$.

□

Theorem 6 shows that the parametric region CR_{IG} can be explored (partitioned) within a finite feasible choice of the parameter $\theta = \theta_0$. This indicates that the above procedure terminates after finite number of partitions of the parameter space.

3.2. Algorithm for multi-parametric non-convex programming problems

The steps of the proposed global optimization framework for a multi-parametric non-convex programming problem with polyhedral constraints are presented as follows:

Step 1: Initialize the upper bound, $\hat{Z}$ of the solution as $\hat{Z} = +\infty$, the optimization variable box $[x^L, x^U]$, the parameter space $[\theta^L, \theta^U]$ and the tolerance value ϵ.

Step 2: Reduce problem (8) into standard non-convex problem by fixing the feasible parameter $\theta = \theta_0$ as:

$$\begin{aligned} Z(\theta_0) \quad &= \quad \min_x \left\{ f_n(x, \theta_0) + \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} b_{ij} x_i x_j \right. \\ &\qquad\qquad \left. + cf(x, \theta_0) \right\} \\ \text{s.t.} \qquad & g(x, \theta_0) \leq 0 \\ & h(x, \theta_0) = 0, \\ & x^L \leq x \leq x^U \end{aligned} \tag{14}$$

Step 3: Solve problem (14) locally within an appropriate optimization variable bound. If the solution is feasible update the current upper bound to the solution as: $C\hat{Z} = \min(\text{current objective value}, \hat{Z})$.

Step 4: Replace each non-convex term in problem (14) by its tight convex under-estimator and solve the resulting problem. If the solution is feasible and $\check{Z}$ is less than $\hat{Z}$ store the solution along with solution set of lower bounds; otherwise fathom the region for $\theta = \theta_0$.

Step 5: Branch (or bisect) the optimization variable having longest side from among those which contribute to the non-convexity of the problem, as:

$$R_1 = \begin{bmatrix} x_1^L & x_1^U \\ x_2^L & x_2^U \\ \vdots & \vdots \\ x_i^L & \frac{(x_i^L+x_i^U)}{2} \\ \vdots & \vdots \\ x_n^L & x_n^U \end{bmatrix}$$

and

$$R_2 = \begin{bmatrix} x_1^L & x_1^U \\ x_2^L & x_2^U \\ \vdots & \vdots \\ \frac{(x_i^L+x_i^U)}{2} & x_i^U \\ \vdots & \vdots \\ x_n^L & x_n^U \end{bmatrix}$$

Step 6: Solve problem (14) inside each of the two subrectangles locally to obtain upper bounds say, $\hat{Z}_2$ and $\hat{Z}_3$. Now compare the obtained upper bounds with previous current upper bound $C\hat{Z}$ and take the minimum one as the current upper bound $C\hat{Z}$.

Step 7: Underestimate every non-convex terms by its tight lower bounding function in each subrectangle R_1 and R_2 as discussed in Subsection 2.1 and solve the resulting convex problems. If the solutions are feasible and less than the current upper bound, store the solutions along with the solution set of lower bounds. Otherwise, fathom the respective rectangle for $\theta = \theta_0$.

Step 8: From the solution set (in Step 7), take the minimum as $\check{Z}$ and compare it with current upper bound $C\hat{Z}$. If the difference $C\hat{Z} - \check{Z} \leq \epsilon$ then go to Step 9. Otherwise, discard $\check{Z}$ from the solution set and go to Step 5 with the sub-rectangle containing the previously found minimum lower bound.

Step 9: Compute M_0, N_0 as discussed in Subsection 2.2 from the respective tighter underestimator subproblem.

Step 10: Characterize the parametric optimal solution $x(\theta)$, Lagrange multipliers $\lambda(\theta)$, $\mu(\theta)$ and the critical region, where the given solution is valid and remove any redundant constraint from this region.

Step 11: Define the rest of the parameter region by reversing one by one the inequalities of the hyperplanes defining the critical region and again remove any occurrence of redundancy as discussed above.

Step 12: For each new region NR_i, set $CR_{IG} = NR_i$ and store each region as: $CR^{\text{rest}} = \bigcup_i NR_i$

Step 13: Compute the Chebyshev center θ_0 and radius r of CR_{IG}. If $r \leq 0$ and CR^{rest} is empty exit; else go to Step 2 with new $\theta = \theta_0$ which is from the rest of the parametric region.

Corollary 2. *Let $CR^{rest} \bigcup CR^Q = X$, $CR^Q \bigcap CR^i = \emptyset\ \forall i$ and $CR^i \bigcap CR^j = \emptyset, \forall i \neq j$ be a partition of X and the difference between $\hat{Z}$ and $\check{Z}$ is a decreasing sequence, then the above algorithm converges.*

Proof. This is an immediate consequence of Theorem 4 and Theorem 6.

□

3.3. Illustrative examples

Example 1

Consider the following multi-parametric non-convex programming problem:

$$\begin{aligned} \min_x \quad & \frac{1}{2}x_1^2 - 2\theta_1 x_1 x_2 + \frac{5}{2}x_2^2 - \theta_1 x_1 - \theta_2 x_1 \\ \text{s.t} \quad & -\frac{1}{3}x_1 + x_2 - 2\theta_2 \leq 0 \\ & -x_1 - \frac{1}{3}x_2 - 2\theta_1 \leq 0 \\ & 1 \leq x_1 \leq 5,\ 1 \leq x_2 \leq 2 \\ & 0 \leq \theta_1 \leq 10,\ 0 \leq \theta_2 \leq 10 \end{aligned} \tag{15}$$

Solving problem (15) using the proposed algorithm for $\epsilon = 10^{-16}$, we have got the following parametric solutions and the union of the corresponding critical regions are depicted in Figure 1 and the algorithm converges within 19.8433 CPU time.

$$CR_1^R = \begin{cases} x(\theta) = \begin{bmatrix} 2.56 - 0.1247\theta_2 - 3.1232\theta_1 \\ 0.5254\theta_1 - 0.0749\theta_2 + 1.5015 \end{bmatrix} \\ 1.5561\theta_1 - 2.0337\theta_2 \leq -0.6765 \\ 0.9483\theta_1 + 0.1497\theta_2 \leq 3.0000 \\ 1.0000\theta_2 \leq 10.0000 \\ -1.0000\theta_1 \leq 0 \end{cases}$$

$$CR_2^R = \begin{cases} x(\theta) = \begin{bmatrix} 1.0532\theta_1 + 7.0670\theta_2 - 23.4361 \\ 26.3377 - 3.9420\theta_2 - 1.7585\theta_1 \end{bmatrix} \\ -2.1061\theta_1 - 8.2747\theta_2 \leq -34.0656 \\ -1.5561\theta_1 + 2.0337\theta_2 \leq 0.6765 \\ 1.0000\theta_1 \leq 10.0000 \end{cases}$$

$$CR_3^R = \begin{cases} x(\theta) = \begin{bmatrix} 0.3798\theta_2 - 1.1057\theta_1 + 5.4361 \\ 10.8632 - 0.3604\theta_2 - 1.1091\theta_2 \end{bmatrix} \\ -0.7443\theta_1 - 2.4857\theta_2 \leq -9.0693 \\ -0.9483\theta_1 - 0.1497\theta_2 \leq -3.0000 \\ 1.5561\theta_1 - 2.0337\theta_2 \leq -0.6765 \\ 1.0000\theta_1 \leq 10.0000 \\ 1.0000\theta_2 \leq 10.0000 \end{cases}$$

$$CR_4^R = \begin{cases} x(\theta) = \begin{bmatrix} 24 - 5.8787\theta_2 - 3.1998\theta_1 \\ 0.0125\theta_1 + 3.6034\theta_2 - 2.5352 \end{bmatrix} \\ 1.0685\theta_1 + 3.5433\theta_2 \leq 10.7465 \\ -1.5561\theta_1 + 2.0337\theta_2 \leq 0.6765 \\ 1.0000\theta_1 \leq 10 \\ -1.0000\theta_1 \leq 0 \\ -1.0000\theta_2 \leq 0 \end{cases}$$

$$CR_5^R = \begin{cases} x(\theta) = \begin{bmatrix} 0.8096\theta_2 - 1.8677\theta_1 + 6.1776 \\ 9.4825 - 0.7760\theta_2 - 1.5458\theta_1 \end{bmatrix} \\ 0.7443\theta_1 + 2.4857\theta_2 \leq 9.0693 \\ -0.9483\theta_1 - 0.1497\theta_2 \leq -3.0000 \\ 1.5561\theta_1 - 2.0337\theta_2 \leq -0.6765 \end{cases}$$

$$CR_6^R = \begin{cases} x(\theta) = \begin{bmatrix} 15.9447 - 2.4510\theta_2 - 1.1321\theta_1 \\ 1.1953\theta_2 - 0.3726\theta_1 + 5.2436 \end{bmatrix} \\ -1.0685\theta_1 - 3.5433\theta_2 \leq -10.7465 \\ 2.1061\theta_1 + 8.2747\theta_2 \leq 34.0656 \\ -1.5561\theta_1 + 2.0337\theta_2 \leq 0.6765 \end{cases}$$

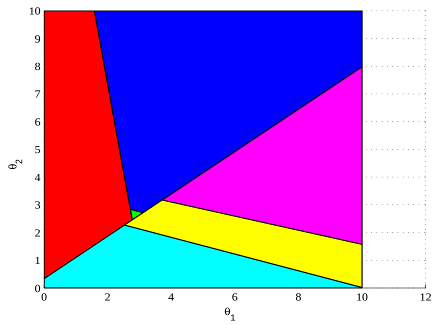

Figure 1. Corresponding critical regions of problem (15)

Example 2

The second test problem is a problem with general non-convex formulation in the objective function and described in the form:

$$\begin{aligned} \min_{x} \quad & x_1 x_2 \sin(x_2) - \frac{x_2}{x_2^2+1} + \theta_1 + \theta_2 \\ \text{s.t} \quad & 5x_1 + x_2 - 2\theta_2 - 2 \leq 0 \\ & x_1 - \frac{1}{3}x_2 - 2\theta_1 - 5 \leq 0 \qquad (16) \\ & -2 \leq \theta_1, \theta_2 \leq 2, \\ & -2 \leq x_1 \leq 2, \ -2 \leq x_2 \leq 4 \end{aligned}$$

Solving problem (16) using the above proposed method for $\epsilon = 10^{-16}$, one can get the following parametric solutions with corresponding critical regions and the total explored parametric region has been shown in Figure 2 and the algorithm converges within 20.6389 CPU time.

$$CR_1^R = \begin{cases} x(\theta) = \begin{bmatrix} 3.1046\mathrm{e}^{-1}\theta_1 + 1.7161\mathrm{e}^{-009}\theta_2 - 0.1507 \\ 1.3964 - 8.8514\mathrm{e}^{-008}\theta_2 - 5.6204\mathrm{e}^{-010}\theta_1 \end{bmatrix} \\ -2.0000\theta_2 \leq 1.3572 \\ 1.0000\theta_1 \leq 2.0000 \\ 1.0000\theta_2 \leq 2.0000 \\ -1.0000\theta_1 \leq 2.0000 \end{cases}$$

$$CR_2^R = \begin{cases} x(\theta) = \begin{bmatrix} 6.4566\mathrm{e}^{-010}\theta_1 + 0.4367\theta_2 + 0.5081 \\ -0.1834\theta_2 - 3.2283\mathrm{e}^{-009}\theta_1 - 0.5403 \end{bmatrix} \\ 2.0000\theta_2 \leq -1.3572 \\ 1.0000\theta_1 \leq 2.0000 \\ -1.0000\theta_1 \leq 2.0000 \\ -1.0000\theta_2 \leq 2.0000 \end{cases}$$

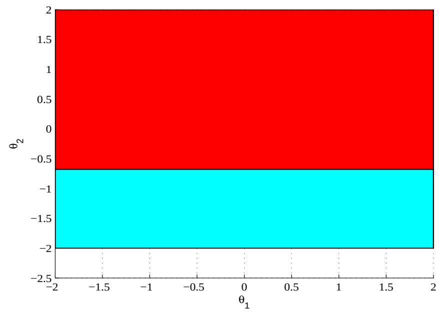

Figure 2. Corresponding critical regions of problem (16)

Example 3

Consider the following problem with generic non-convex formulation appeared on its objective function:

$$\begin{aligned} \min_{x} f_2 = \cos(x_1)\sin(x_2) - \frac{x_1\theta_1}{x_2^2+1} & \\ \text{s.t} \quad -\frac{1}{3}x_1 + x_2 - 2\theta_2 \leq 0 & \qquad (17) \\ -x_1 - \frac{1}{3}x_2 - 2\theta_1 \leq 0 & \\ -1 \leq x_1 \leq 2, \ -1 \leq x_2 \leq 1 & \\ 0 \leq \theta_1, \theta_2 \leq 2 & \end{aligned}$$

After solving problem (17) using the algorithm described in this paper, we have got a parametric solution given below. The corresponding critical region is shown in Figure 3. For this problem, the algorithm converges within tolerance error $\epsilon = 10^{-16}$ and 42.5103 CPU time.

$$
CR_1^R = \begin{cases} x(\theta) = \begin{bmatrix} 1.0414 \times 10^{-8}\theta_2 - 7.02640\theta_1 + 7.3198 \\ 3.4634 - 1.0399 \times 10^{-8}\theta_2 - 3.46340\theta_1 \end{bmatrix} \\ -1.14460\theta_1 - 2.00000\theta_2 \leq -1.0479 \\ 6.18030\theta_1 \leq 8.4732 \\ 1.00000\theta_2 \leq 2.0000 \\ -1.00000\theta_1 \leq 0 \\ -1.00000\theta_2 \leq 0 \end{cases}
$$

$$
CR_2^R = \begin{cases} x(\theta) = \begin{bmatrix} 1.1239 \times 10^{-8}\theta_2 - 1.70820\theta_1 + 0.8636 \\ 0.856310\theta_2 - 2.22580\theta_1 + 0.2011 \end{bmatrix} \\ 1.14460\theta_1 + 2.00000\theta_2 \leq 1.0479 \\ -1.00000\theta_1 \leq 0 \\ -1.00000\theta_2 \leq 0 \end{cases}
$$

$$
CR_3^R = \begin{cases} x(\theta) = \begin{bmatrix} 18.7304 - 1.3873 \times 10^{-9}\theta_2 - 10.92320\theta_1 \\ 5.3693 - 3.478 \times 10^{-9}\theta_2 - 3.20330\theta_1 \end{bmatrix} \\ 0.41910\theta_1 - 2.00000\theta_2 \leq 0.8118 \\ -6.18030\theta_1 \leq -8.4732 \\ 1.00000\theta_1 \leq 2.0000 \\ 1.00000\theta_2 \leq 2.0000 \\ -1.00000\theta_2 \leq 0 \end{cases}
$$

$$
CR_4^R = \begin{cases} x(\theta) = \begin{bmatrix} 26.2252 - 3.20657 \times 10^{-6}\theta_2 - 12.99230\theta_1 \\ 6.3876 - 3.1265 \times 10^{-6}\theta_2 - 3.20330\theta_1 \end{bmatrix} \\ -0.41910\theta_1 + 2.00000\theta_2 \leq -0.8118 \\ 1.00000\theta_1 \leq 2.0000 \\ -1.00000\theta_2 \leq 0 \end{cases}
$$

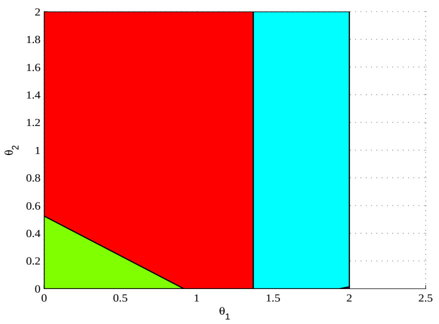

Figure 3. Corresponding critical regions of problem (17)

4. Conclusion

In this paper we have given a detailed description of an approximate solution algorithm for multi-parametric non-convex programming problems with convex polyhedron constraints. The approach is constructed through successive convex relaxation of each of the non-convex terms followed by employing sensitivity analysis theory. Special attention is given to general non-convex formulations in the objective function with convex polyhedral constraints. The algorithm has been tested for a variety of example problems having a general non-convex formulation at their objective functions. The proposed algorithm can be extended to a general but twice continuously differentiable nonlinear multi-parametric programming models by making some mathematical modifications on the presented algorithm to transform nonlinear constraints into the objective part.

Acknowledgments

This work is in-part supported by the Swedish International Science Program (ISP), through the project at the Department of Mathematics, Addis Ababa University. The authors would like to thank the two anonymous reviewers for their valuable comments and suggestions to improve the earlier version of the manuscript.

References

[1] Li Z. and Ierapetritou G. M., A New Methodology for the General Multiparametric Mixed-Integer Linear Programming (MILP) Problems, *Industrial & Engineering Chemistry Research*, 46, 5141-5151 (2007).

[2] Dua V. Bozinis A. N. and Pistikopoulos N. E., A multi-parametric Programming Approach for mixed-integer quadratic engineering problems, *Computers & Chemical Engineering*, 26, 715-733 (2002).

[3] Pistikopous, N. E., Georgiads C. M. and Dua V., (Editors) Multiparametric programming: Theory, algorithm, and application, WILEY-VCH Verlag GmbH and Co. KGaA, (2007).

[4] Dua V., Pistikopoulos, E. N., An algorithm for the solution of multiparametric mixed integer linear programming problems, *Annals of Operations Research*, 99, 123 - 139 (2001).

[5] Faísca P. N, Dua V., Rustem B., Saraiva M. P., Pistikopoulos N. E., Parametric global optimisation for bilevel programming, *Journal of Global Optimization*, 38, 609-623 (2006).

[6] Faísca P. N., Saraiva M. P., Rustem B., Pistikopoulos N. E. A multiparametric programming approach for multilevel hierarchical and decentralized optimization problems, *Computational Management Science*, 6, 377-397 (2009).

[7] Tøndel P., Johansen T. A. and Bemporad A., Further Results on Multiparametric Quadratic Programming, *in Proceedings of 42nd IEEE Conference on Decision and Control*, 3, 3173-3178 (2003).

[8] Al-Khayyal A. F., Jointly constrained bilinear programms and related problems: An overview, *Computers Math. applic*, 19, 53-62 (1990).

[9] Adjiman S. C., Dallwing S., Floudas A. C., Neumaier A., A global optimization method, αBB, for general twice-differentiable constrained NLPs I.Theoretical advances, *Computers and Chemical Engineering*, 22, 1137 - 1158 (1998).

[10] Androulakis P. I., Maranas D. C., and Floudas A. C., αBB: A Global Optimization Method for General Constrained Nonconvex Problems, *Journal of Global Optimization*, 7, 337-363 (1995).

[11] Fiacco V. A., Sensitivity analysis for nonlinear programming using penalty methods, *Mathematical Programming*, 10, 287-311 (1976).

[12] Fiacco V. A., Introduction to Sensitivity and Stability Analysis in Nonlinear Programming, Acadamic press, (1983).

Abay Molla Kassa *is a lecturer in the department of Chemical Engineering at Addis Ababa University (AAiT). He received Bachelors degree in applied Mathematics from University of Gondar, Ethiopia, and his Masters degree in Applied Mathematics from Addis Ababa University, Ethiopia.*

Semu Mitiku Kassa *is an Associate Professor in the department of Mathematics at Addis Ababa University. He received Bachelors and Masters degrees in Mathematics from Addis Ababa University, Ethiopia, and a PhD degree in Operations Research from University of Karlsruhe, Germany.*