A parametric method to solve quadratic programming problems with fuzzy costs

Ricardo C. Silva[1] Carlos Cruz[2] Akebo Yamakami[1]

1.Department of Telematics, School of Electrical and Computer Engineering,
University of Campinas, P.O.Box 6101, 13083-970, Campinas-SP, Brazil.
Email: {rcoelhos,akebo}@dt.fee.unicamp.br
2.Department of Computer Science and Artificial Intelligence,
University of Granada, E-18071, Granada, Spain
Email: carloscruz@decsai.ugr.es

Abstract— *This work describes a novel fuzzy-sets-based method to solve a particular class of quadratic programming problems which have vagueness coefficients in the objective function. Quadratic programming problems are of utmost importance in an increasing variety of practical fields. In addition, as the ambiguity and vagueness are natural and ever-present in real-life situations requiring solutions, it makes perfect sense to attempt to address them using fuzzy quadratic programming problems. Also, two other methods to solve this kind of problems are briefly described. The proposal uses two phases to solve fuzzy quadratic programming problems. In the first, phase we parametrize the fuzzy problem in several classical alpha-problems with different cutting levels. In the second, phase each of these alpha-problems is solved by using conventional solving techniques. The final fuzzy solution to the problem is obtained by integrating all of these particular alpha-solutions. The results obtained using these two methods are compared with the two-phased proposal outlined above.*

Keywords— Fuzzy sets, decision making, fuzzy mathematical programming, quadratic optimization.

1 Introduction

The conception of Artificial Intelligence (AI) in the early fifties was inspired by the wide variety of physical and mental tasks performed by humans without any measurements and any computations. Thus, this forward thinking described the definition of AI as it is widely known: “the study and design of intelligent agents [14]”, where an intelligent agent is a system that perceives its environment and takes actions which maximize its chances of success[15]. The capability to compute and reason with perception-based information can be applied to real-world problems in which decision-relevant information is a mixture of measurements and perceptions [20], where in general, measurements are crisp, although perceptions are fuzzy.

The literature shows that the best way of modeling these types of problems is using Soft Computing methodologies. According to Verdegay et al.[17], Soft Computing is a family of problem-resolution methods headed by approximate reasoning, functional and optimization approximation methods, which now include search methods. More specifically, the search methods use thorough evaluations to find regions with approximate points of optimal solutions that are obtained by using optimization methods.

These methods, which are algorithms based on mathematical knowledge, are used to guarantee the convergence to the optimal solution. They are covered by the area of mathematical programming. This area has several sub-fields where the quadratic programming is a special type that optimizes a quadratic objective function of several variables subject to linear constraints. A precise definition of the constraints and the objective function to be optimized is necessary to obtain exact optimal solutions of quadratic programming problems.

Nevertheless, in most real practical applications (portfolio, game theory, engineering modeling, design and control, logistics, etc.) one lacks this kind of exact knowledge, and only approximate, vague and imprecise values are known. Moreover, these imprecise values can be dealt with fuzzy logic. In this case, the concept of fuzzy mathematical programming emerges when it is used.

With this in mind, the objective of this paper is to review some methods that are related to quadratic topic and to outline a soft computing model used to solve novel fuzzy quadratic programming problems.

The paper is organized as follows: Section 2 briefly introduces the different approaches to solving the quadratic programming problem with fuzzy costs. In this section, a novel approach is developed to solve quadratic programming problems with fuzzy costs where these are transformed into parametrical quadratic multiobjective programming problems. To clarify the above developments, three numerical examples are analyzed in section 3. Finally, conclusions are presented in Section 4.

2 Quadratic optimization with fuzzy costs

An optimization problem that is described with a quadratic objective function subject to linear constraints is called a “Quadratic Programming” problem. QP can be viewed both as a special case of the nonlinear programming and a generalization of the linear programming. Several applications and methods can be found in [3, 8, 9, 16, 19]. A quadratic programming problem can be formulated as

$$\begin{array}{ll} \min & \mathbf{c}^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\mathbf{Q}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0 \end{array} \qquad (1)$$

where $\mathbf{x}$ and $\mathbf{c}$ are n vectores of real numbers, $\mathbf{Q}$ is an $n \times n$ symmetric matrix of real numbers, $\mathbf{A}$ is an $m \times n$ matrix of real numbers and $\mathbf{b}$ is an m vector of real numbers.

ISBN: 978-989-95079-6-8

In this work, quadratic programming problems with fuzzy costs are considered because there are cases of real world problems whose parameters are seldom known exactly and have to be estimated by the decision maker. Hence, the $n \times n$ symmetric matrix $\mathbf{Q}$ and the n vector $\mathbf{c}$ have fuzzy numbers in these components. The application of fuzzy logic is also a way to describe, mathematically, this vagueness as described in [2, 10, 13, 18]. The uncertainties of costs of the objective function can be dealt with by fuzzy numbers.

Thus, this set of problems with uncertain costs can be formalized in the following form:

$$\begin{array}{ll} \min & \tilde{\mathbf{c}}^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\tilde{\mathbf{Q}}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq 0 \end{array} \qquad (2)$$

where the fuzzy numbers are characterized by membership functions that are defined by decision makers. The membership functions can be defined as

$$\mu_j, \mu_{ij} : \mathbb{R} \to [0,1], \ \ i,j \in \mathbb{J} = \{1,2,\ldots,n\}$$

In particular these membership functions will be supposed as:

$$\mu_j(y) = \begin{cases} 0 & \text{if } c_j^U \leq y \text{ or } y \leq c_j^L \\ h_j(y) & \text{if } c_j^L \leq y \leq c_j \\ g_j(y) & \text{if } c_j \leq y \leq c_j^U \end{cases} \quad j \in \mathbb{J} \qquad (3)$$

and

$$\mu_{ij}(y) = \begin{cases} 0 & \text{if } q_{ij}^U \leq y \text{ or } y \leq q_{ij}^L \\ h_{ij}(y) & \text{if } q_{ij}^L \leq y \leq q_{ij} \\ g_{ij}(y) & \text{if } q_{ij} \leq y \leq q_{ij}^U \end{cases} \quad i,j \in \mathbb{J} \qquad (4)$$

where $\mathbf{h}(\cdot)$ and $\mathbf{g}(\cdot)$ are assumed to be strictly increasing and decreasing continuous functions, respectively, $h_j(c_j) = g_j(c_j) = 1, \ j \in \mathbb{J}$ and $h_{ij}(q_{ij}) = g_{ij}(q_{ij}) = 1, \ i,j \in \mathbb{J}$.

Bellman and Zadeh describe the *fuzzy decision* in [4] as the intersection of goals and constraints which is formalized in the following definition:

Definition 1 *Assume that we are given a fuzzy goal μ_G and a fuzzy constraint μ_C in a space of alternative X. Then, μ_G and μ_C combine to form a* fuzzy decision*, μ_D, which is a fuzzy set resulting from intersection of μ_G and μ_C. In symbols*

$$\sup_{\mathbf{x}\in X} \mu_D(\mathbf{x}) = \sup_{\mathbf{x}\in X} [\mu_G(\mathbf{x}) \wedge \mu_C(\mathbf{x})]$$

Note that in Definition 1, the goals and the constraints enter into the expression for μ_D in the same way. Thus, it is able to find a maximizing decision to an extremum problem for a scalar function, as described in [13]. Let ϕ be denoted by $\phi : [0,1] \to [0,1]$ that implies $\phi(\alpha) = \sup_{\mathbf{x}\in\mathbf{X}(\alpha)} \mu_G(\mathbf{x})$ where $X(\alpha) = \{\mathbf{x} \in \mathbb{R}^n \mid \mu_X(\mathbf{x}) \geq \alpha\}$. If ϕ is continuous in [0,1] then ϕ has a *fix point* $\overline{\alpha}$ and, therefore,

$$\sup_{\mathbf{x}\in X} \mu_D(\mathbf{x}) = \sup_{\mathbf{x}\in X(\overline{\alpha})} \mu_G(\mathbf{x}) = \overline{\alpha}$$

.

A novel parametric approach to solve quadratic programming problems with fuzzy costs will be presented in the next sub-section. This approach transforms the original problem into a parametric multi-objective quadratic programming problem.

2.1 *Multi-objective approach*

A multi-objective approach to solve a linear programming problem with imprecise costs is described in [6, 7]. As the linear problem is a particular case of quadratic problem, this approach can be extended to solve quadratic programming problems with fuzzy costs.

The quadratic objective function can be divided into two parts, where the first one is a linear term and the second one is a quadratic term. According to this, the fuzzy costs can only be in the first part or only the second part or in both. In this section, these three ways will be presented separately.

The linear problem considered in [6] used trapezoid membership functions for the costs but here, for the sake of simplicity, they will be supposed to be like (3) and (4).

Then, by considering the $(1-\alpha)$-cut of every cost, $\alpha \in [0,1]$,

$$\begin{array}{l} \forall\, x \in \mathbb{R}, \ \ \mu_j(x) \geq 1-\alpha \Leftrightarrow h_j^{-1}(1-\alpha) \leq x \leq g_j^{-1}(1-\alpha), \\ \forall\, x \in \mathbb{R}, \ \ \mu_{ij}(x) \geq 1-\alpha \Leftrightarrow h_{ij}^{-1}(1-\alpha) \leq x \leq g_{ij}^{-1}(1-\alpha), \end{array}$$

where $i,j \in \mathbb{J} = \{1,2,\ldots,n\}$.

2.1.1 *Quadratic programming problem with fuzzy costs vector of linear term*

The approaches described in this section solve a quadratic programming problem with a vector $\tilde{\mathbf{c}}$ of fuzzy numbers. This vector describes the costs of the objective function and Problem 2 can be rewritten as follows:

$$\begin{array}{ll} \min & \tilde{\mathbf{c}}^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\mathbf{Q}\mathbf{x} \\ \text{s.t.} & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{array} \qquad (5)$$

Thus, according to some results by Bitran [5], it has been shown that a fuzzy solution to (5) may be found from the parametric solution of the multi-objective parametric quadratic programming problem

$$\begin{array}{ll} \min & \big[(\mathbf{c^1})^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\mathbf{Q}\mathbf{x}, (\mathbf{c^2})^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\mathbf{Q}\mathbf{x}, \ldots, \\ & \qquad\qquad (\mathbf{c^{2^n}})^{\mathbf{t}}\mathbf{x} + \frac{1}{2}\mathbf{x}^{\mathbf{t}}\mathbf{Q}\mathbf{x}\big] \\ \text{s.t.} & \mathbf{A}\mathbf{x} \leq \mathbf{b}, \ \ \mathbf{x} \geq \mathbf{0}, \\ & \mathbf{c^k} \in \mathbb{E}(1-\alpha), \ \ \alpha \in [0,1], \ \ k = 1,2,\ldots,2^n, \end{array} \qquad (6)$$

where $\mathbb{E}(1-\alpha)$, for each $\alpha \in [0,1]$, is the set of vectors in $\mathbb{R}^n$ such that each of its components is either in the lower bound, $h_j^{-1}(1-\alpha)$, or in the upper bound, $g_j^{-1}(1-\alpha)$, of the respective $(1-\alpha)$-cut, that is, $\forall\, k = 1,2,\ldots,2^n$, and $\forall\, j \in \mathbb{J}$

$$c^k = (c_1^k, c_2^k, \ldots, c_n^k) \in \mathbb{E}(1-\alpha) \Leftrightarrow c_j^k = \begin{cases} h_j^{-1}(1-\alpha) & \text{or} \\ g_j^{-1}(1-\alpha). \end{cases}$$

Now observe that to find a parametric solution to (6), from which one can obtain a fuzzy solution to (5), one may use any classical multiobjective quadratic programming approach.

2.1.2 *Quadratic programming problem with fuzzy costs matrix of quadratic term*

The approach described in this section solves a quadratic programming problem with matrix $\tilde{\mathbf{Q}}$ of fuzzy numbers. This

ISBN: 978-989-95079-6-8

vector describes the costs of the objective function and Problem 2 can be rewritten as follows

$$\begin{aligned} \min \quad & \mathbf{c^t x} + \tfrac{1}{2}\mathbf{x^t}\tilde{\mathbf{Q}}\mathbf{x} \\ \text{s.t.} \quad & \mathbf{Ax} \le \mathbf{b} \\ & \mathbf{x} \ge \mathbf{0}. \end{aligned} \tag{7}$$

As it was shown above, a fuzzy solution to (7) may be found from the parametric solution of the multiobjective parametric quadratic programming problem

$$\begin{aligned} \min \quad & \left[\mathbf{c^t x} + \tfrac{1}{2}\mathbf{x^t Q^1 x}, \mathbf{c^t x} + \tfrac{1}{2}\mathbf{x^t Q^2 x}, \dots, \mathbf{c^t x} + \tfrac{1}{2}\mathbf{x^t Q^{2^{n^2}} x}\right] \\ \text{s.t.} \quad & \mathbf{Ax} \le \mathbf{b},\ \ \mathbf{x} \ge \mathbf{0}, \\ & \mathbf{Q^k} \in \mathbb{E}(1-\alpha),\ \ \alpha \in [0,1],\ \ k = 1,2,\dots,2^{n^2}, \end{aligned} \tag{8}$$

where $\mathbb{E}(1-\alpha)$, for each $\alpha \in [0,1]$, is the set of vectors in $\mathbb{R}^n$ such that each of its components is either in the lower bound, $h_{ij}^{-1}(1-\alpha)$, or in the upper bound, $g_{ij}^{-1}(1-\alpha)$, of the respective $(1-\alpha)$-cut, that is, $\forall\, k = 1,2,\dots,2^{n^2}$, and $\forall\, i,j \in \mathbb{J}$

$$\mathbf{Q^k} = (q_{11}^k, \dots, q_{1n}^k, \dots, q_{nn}^k) \in \mathbb{E}(1-\alpha) \Leftrightarrow$$
$$\Leftrightarrow q_{ij}^k = \begin{cases} h_{ij}^{-1}(1-\alpha) & \text{or} \\ g_{ij}^{-1}(1-\alpha). \end{cases}$$

Now observe that to find a parametric solution to (8), from which one can obtain a fuzzy solution to (7), one may use any classical multiobjective quadratic programming approach.

2.1.3 *Quadratic programming problem with all fuzzy costs*

Thus, according to the parametric transformations shown above, a fuzzy solution to (2) may be found from the parametric solution of the multiobjective parametric quadratic programming problem

$$\begin{aligned} \min \quad & \Big[(\mathbf{c^1})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^1 x}, (\mathbf{c^2})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^1 x}, \dots, \\ & \dots, (\mathbf{c^{2^n}})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^1 x}, (\mathbf{c^1})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^2 x}, \dots, \\ & \dots, (\mathbf{c^{2^n}})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^2 x}, \dots, (\mathbf{c^{2^n}})^\mathbf{t}\mathbf{x} + \tfrac{1}{2}\mathbf{x^t Q^{2^{n^2}} x}\Big] \\ \text{s.t.} \quad & \mathbf{Ax} \le \mathbf{b},\ \ \mathbf{x} \ge \mathbf{0}, \\ & \mathbf{c^k}, \mathbf{Q^p} \in \mathbb{E}(1-\alpha),\ \ \alpha \in [0,1], \\ & k = 1,2,\dots,2^n \text{ and } p = 1,2,\dots,2^{n^2}, \end{aligned} \tag{9}$$

where $\mathbb{E}(1-\alpha)$, for each $\alpha \in [0,1]$, is the set of vectors in $\mathbb{R}^n$ such that each of its components is either in the lower bound, $h_j^{-1}(1-\alpha)$, or in the upper bound, $g_j^{-1}(1-\alpha)$, of the respective $(1-\alpha)$-cut, that is, $\forall\, k = 1,2,\dots,2^n$, and $\forall\, i,j \in \mathbb{J}$

$$c^k = (c_1^k, c_2^k, \dots, c_n^k) \in \mathbb{E}(1-\alpha) \Leftrightarrow c_j^k = \begin{cases} h_j^{-1}(1-\alpha) & \text{or} \\ g_j^{-1}(1-\alpha). \end{cases}$$

and

$$\mathbf{Q^k} = (q_{11}^k, \dots, q_{1n}^k, \dots, q_{nn}^k) \in \mathbb{E}(1-\alpha) \Leftrightarrow$$
$$\Leftrightarrow q_{ij}^k = \begin{cases} h_{ij}^{-1}(1-\alpha) & \text{or} \\ g_{ij}^{-1}(1-\alpha). \end{cases}$$

The obtained parametrical solutions for any of multi-objective models above, to the different α values, generate a set of solutions and then we use the Representation Theorem to integrate all of these particular alpha-solutions.

2.2 *Liu's approach*

Transforming a fuzzy quadratic programming problem into a two-level mathematical programming problem for finding the bounds of the fuzzy values in the objective function and in the set of constraints is discussed in [11, 12]. These papers describe how to transform the two-level mathematical program into the conventional one-level quadratic program. However, such a method is also valid when only the objective is fuzzy, that is, when only the vector of costs are fuzzy, and for this reason it shall be considered here. According to the goal of this work, the coefficient **c** of Problem (5) is fuzzy.

The authors derive the membership function of the goal, and then they apply Zadeh's extension principle to transform the fuzzy quadratic problem into family of classical quadratic problems that can be solved by conventional optimization techniques. Thus, the membership function of the objective function can be defined as

$$\mu_{\tilde{Z}}(z) = \sup_{\mathbf{c}} \min\{\mu_{\tilde{c}_j}(c_j),\ \forall j | z = Z(\mathbf{c})\} \tag{10}$$

where $Z(\mathbf{c})$ is the goal of the conventional quadratic problem. Membership function $\mu_{\tilde{Z}}$ can be computed by finding the functions that describe the shape of the left and right sides of the fuzzy numbers. Then, it is possible to obtain the upper bound of the objective value Z_α^U and your lower bound Z_α^L to each value α. Thus, Z_α^U is the maximum and Z_α^L is the minimum of $Z(\mathbf{c})$, respectively, that can be described as

$$Z_\alpha^U = \max_{c_j} \left\{ \begin{array}{ll} \min_{\mathbf{x}} & \sum_{j=1}^n c_j x_j + \frac{1}{2}\sum_{j=1}^n \sum_{l=1}^n q_{jl} x_j x_l \\ \text{s.a} & \sum_{j=1}^n a_{ij} x_j \le b_i,\ \ i = 1,\dots,m \\ & x_j \ge 0,\ \ j = 1,\dots,n \end{array} \right. \tag{11}$$

$$Z_\alpha^L = \min_{c_j} \left\{ \begin{array}{ll} \min_{\mathbf{x}} & \sum_{j=1}^n c_j x_j + \frac{1}{2}\sum_{j=1}^n \sum_{k=1}^n q_{jk} x_j x_k \\ \text{s.a} & \sum_{j=1}^n a_{ij} x_j \le b_i,\ \ i = 1,\dots,m \\ & x_j \ge 0,\ \ j = 1,\dots,n \end{array} \right. \tag{12}$$

where $c_j \in [(c_j)_\alpha^L, (c_j)_\alpha^U]$, for any $\alpha \in [0,1]$ chosen by decision maker.

These two formulations above are solved by outer-level and inner-level programs, respectively. The outer-level program obtains the values c_j that are used with parameters by inner-level program. The inner-level program solves a classical quadratic programming problem with the data obtained by outer-level program. The authors state that the formulation of two-level quadratic problems is a generalization of the conventional parametrical quadratic programming problem. Thus, the first two-level mathematical program can be transformed into the following quadratic problem by dual formulation:

$$\begin{aligned} \max \quad & \textstyle\sum_{i=1}^m b_i \lambda_i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n q_{ij} x_i x_j \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^m a_{ij}\lambda_i + \delta_j - \sum_{i=1}^n q_{ij} x_i = c_j,\ \ j = 1,2,\dots,n \\ & (c_j)_\alpha^L \le c_j \le (c_j)_\alpha^U,\ \ j = 1,2,\dots,n \\ & \lambda_i, \delta_j \ge 0,\ \ i = 1,2,\dots,m \text{ and } j = 1,2,\dots,n. \end{aligned} \tag{13}$$

Since both the inner-level program and the outer-level of the second program have the same minimization operation, they can be combined into a conventional one-level program with the constraints of the two programs considered simultaneously. Consequently, some points must be analyzed which

ISBN: 978-989-95079-6-8

are shown in [11, 12]. The second program can be described as

$$\begin{array}{rl} \max & \sum_{j=1}^{n}(c_j)_{\alpha}^{L} + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} q_{ij}x_i x_j \\ \text{s.t.} & \sum_{i=1}^{n} a_{ij}x_i \leq b_i,\ \ i=1,2,\ldots,m \\ & x_j \geq 0,\ \ j=1,2,\ldots,n \end{array} \tag{14}$$

2.3 Ammar's approach

A method to solve a quadratic programming problem with fuzzy coefficients in the objective function and in the set of constraint is described in [1]. However, such a method is also valid when only the objective is fuzzy, that is, when only the matrix of costs are fuzzy, and for this reason it shall be considered here. This approach can be formulated in the following way:

$$\begin{array}{rl} \min & \mathbf{x^t}\tilde{\mathbf{Q}}\mathbf{x} \\ \text{s.t.} & \mathbf{Ax} \leq \mathbf{b} \\ & \mathbf{x} \geq \mathbf{0}. \end{array} \tag{15}$$

where all decision variables are non negatives and the linear term in objective function is not used. However, in this work only costs of the objective functions are imprecise.

Thus, Problem (15) can be reformulated by using α-cutting levels as

$$\begin{array}{rrl} (P_\alpha): & \min & \mathbf{x^t}[\mathbf{Q}_\alpha^-, \mathbf{Q}_\alpha^+]\mathbf{x} \\ & \text{s.a} & \mathbf{Ax} \leq \mathbf{b} \\ & & \mathbf{x} \geq \mathbf{0}. \end{array} \tag{16}$$

where $\alpha \in (0,1]$. $\mathbf{Q}_\alpha^-$ represents a matrix with the lower values of the interval and $\mathbf{Q}_\alpha^+$ represents a matrix with the upper values of the interval.

[1] describes how Problem (16) can be written in two quadratic problems. Then, these problems are formulated as follows

$$\begin{array}{rrl} (P_\alpha^-): & \min & \mathbf{x^t}\mathbf{Q}_\alpha^-\mathbf{x} \\ & \text{s.a} & \mathbf{Ax} \leq \mathbf{b} \\ & & \mathbf{x} \geq \mathbf{0}. \end{array} \tag{17}$$

$$\begin{array}{rrl} (P_\alpha^+): & \min & \mathbf{x^t}\mathbf{Q}_\alpha^+\mathbf{x} \\ & \text{s.t.} & \mathbf{Ax} \leq \mathbf{b} \\ & & \mathbf{x} \geq \mathbf{0}. \end{array} \tag{18}$$

where the first problem uses the lower bound of the interval of the α-cutting level and the second problem uses the upper bound.

The two quadratic problems (17) and (18) can be solved by using the conditions of Karush-Kuhn-Tucker's optimality to each α value. The optimal solution of the original problem (16), (P_α), is inside the interval formed by the optimal solutions of each one problems (P_α^-) and (P_α^+).

3 Numerical example

In this section, we illustrate a quadratic programming problem that will be solved with three different way. Each way is represented by different fuzzy environment in each problem, i.e., the first problem has fuzzy costs in the linear term, while the second has fuzzy costs in the quadratic term, and the third has fuzzy costs in all coefficients in the objective function.

Example 1 *Consider the following quadratic programming problem with fuzzy coefficients in the linear term:*

$$\begin{array}{rl} \min & (-6,-5,-4)x_1 + (1,1.5,2)x_2 + 2x_1^2 - 2x_1x_2 + x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & 2x_1 - x_2 \leq 4 \\ & x_1, x_2 \geq 0. \end{array} \tag{19}$$

The membership functions are:

$$\mu_{-5}(c_1) = \begin{cases} c_1 + 6 & -6 \leq c_1 \leq -5 \\ -c_1 - 4 & -5 \leq c_1 \leq -4 \end{cases}$$

$$\mu_{1.5}(c_2) = \begin{cases} 2(c_2 - 1) & 1 \leq c_2 \leq 1.5 \\ 2(2 - c_2) & 1.5 \leq c_2 \leq 2 \end{cases}$$

From the membership function for this costs an interval representation according to (3) can be given:

$$c_1^{0.2} = [-5.8, -4.2], \quad c_1^{0.4} = [-5.6, -4.4],$$
$$c_1^{0.6} = [-5.4, -4.6], \quad c_1^{0.8} = [-5.2, -4.8]$$

$$c_2^{0.2} = [1.1, 1.9], \quad c_2^{0.4} = [1.2, 1.8],$$
$$c_2^{0.6} = [1.3, 1.7], \quad c_2^{0.8} = [1.4, 1.6]$$

with $c_1^0 = [-6,-4]$, $c_1^1 = [-5,-5]$ *and* $c_2^0 = [1,2]$, $c_2^1 = [1.5, 1.5]$*, and now* $\mathbb{M} = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$.

Thus, Problem (19) can be re-formulated as:

$$\begin{array}{rl} \min & (\alpha-6)x_1 + \frac{1}{2}(\alpha+2)x_2 + 2x_1^2 - 2x_1x_2 + x_2^2 \\ \min & (\alpha-6)x_1 + \frac{1}{2}(4-\alpha)x_2 + 2x_1^2 - 2x_1x_2 + x_2^2 \\ \min & (-\alpha-4)x_1 + \frac{1}{2}(\alpha+2)x_2 + 2x_1^2 - 2x_1x_2 + x_2^2 \\ \min & (-\alpha-4)x_1 + \frac{1}{2}(4-\alpha)x_2 + 2x_1^2 - 2x_1x_2 + x_2^2 \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & 2x_1 - x_2 \leq 4 \\ & x_1, x_2 \geq 0. \end{array} \tag{20}$$

Problem (20) is solved by weighting objectives, with $\alpha = 0.8$*, where an auxiliary problem can be obtained in the following way:*

$$\begin{array}{rl} \min & \omega_1[-5.2x_1 + 1.4x_2] + \omega_2[-5.2x_1 + 1.6x_2] + \\ & +\omega_3[-4.8x_1 + 1.4x_2] + \omega_4[-4.8x_1 + 1.6x_2] + \\ & +2x_1^2 - 2x_1x_2 + x_2^2 \\ s.t. & x_1 + x_2 \leq 2,\ \ 2x_1 - x_2 \leq 4,\ \ x_1, x_2 \geq 0. \end{array} \tag{21}$$

Different optimal solutions to each of the weighting vectors is obtained by choosing the canonical base of $\mathbb{R}^4$ *as weighting vectors, as it is presented in Table 1.*

Table 1: Result of Problem 21 to each vector ω.

ω	x_1	x_2	FunObj
(1,0,0,0)	1.4600	0.5400	-3.8580
(0,1,0,0)	1.4800	0.5200	-3.7520
(0,0,1,0)	1.4200	0.5800	-3.2820
(0,0,0,1)	1.4400	0.5600	-3.1680

By taking $\omega_1 = \omega_2 = \omega_3 = \omega_4 = \frac{1}{4}$ *and solving (21) the optimal solution is* $x^* = [1.45, 0.55]$ *and value of objective function is -3.5125.*

On the other hand, according to the method described in 2.2, Problem (19), which is the outer-level program, can be

ISBN: 978-989-95079-6-8

rewritten as:

$$\begin{array}{ll} \max & 2\lambda_1 + 4\lambda_2 - \frac{1}{2}(4x_1^2 - 4x_1x_2 + 2x_2^2) \\ \text{s.t.} & \lambda_1 + 2\lambda_2 - \delta_1 - 4x_1 + 2x_2 = c_1 \\ & \lambda_1 - 1\lambda_2 - \delta_2 + 2x_1 - 2x_2 = c_2 \\ & -5.2 \leq c_1 \leq -4.8 \\ & 1.4 \leq c_2 \leq 1.6 \\ & x_1, x_2, \lambda_1, \lambda_2, \delta_1, \delta_2 \geq 0. \end{array} \quad (22)$$

and the inner-level program can be written as:

$$\begin{array}{ll} \min & -5.2x_1 + 1.4x_2 + \frac{1}{2}(4x_1^2 - 4x_1x_2 + 2x_2^2) \\ \text{s.t.} & x_1 + x_2 \leq 2 \\ & 2x_1 - x_2 \leq 4 \\ & x_1, x_2 \geq 0. \end{array} \quad (23)$$

This model is a traditional quadratic program that can represent the two models described in 2.2. Hence, by solving the outer-level program the optimal solution is $x^* = [1.3, 0]$ *and value of objective function is -3.38. On the other hand, by solving the inner-level program the optimal solution is* $x^* = [1.45, 0.55]$ *and value of objective function is -3.8575.*

Example 2 *Consider the following quadratic programming problem with fuzzy coefficients in the quadratic term:*

$$\begin{array}{ll} \min & -5x_1 + 1.5x_2 + (1,2,3)x_1^2 + (1,2,3)x_1x_2 + (0,1,2)x_2^2 \\ \text{s.t.} & x_1 + x_2 \geq 2 \\ & 2x_1 - x_2 \geq 4 \\ & x_1, x_2 \geq 0. \end{array} \quad (24)$$

The membership functions are

$$\mu_2(c_1) = \begin{cases} c_1 - 1 & 1 \leq c_1 \leq 2 \\ 2 - c_1 & 2 \leq c_1 \leq 3 \end{cases}$$

$$\mu_1(c_2) = \begin{cases} c_2 & 0 \leq c_2 \leq 1 \\ 2 - c_2 & 1 \leq c_2 \leq 2 \end{cases}$$

From the membership function for these costs an interval representation according to (4) can be given:

$$c_1^{0.2} = [1.2, 2.8], \quad c_1^{0.4} = [1.4, 2.6], \quad c_1^{0.6} = [1.6, 2.4], \quad c_1^{0.8} = [1.8, 2.2]$$

$$c_2^{0.2} = [0.2, 1.8], \quad c_2^{0.4} = [0.4, 1.6], \quad c_2^{0.6} = [0.6, 1.4], \quad c_2^{0.8} = [0.8, 1.2]$$

with $c_1^0 = [1,3]$, $c_1^1 = [2,2]$ *and* $c_2^0 = [0,2]$, $c_2^1 = [1,1]$, *and now* $\mathbb{M} = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$.

Thus, Problem (24) can be re-formulated as

$$\begin{array}{ll} \min & -5x_1 + 1.5x_2 + (\alpha+1)x_1^2 - (\alpha+1)x_1x_2 + \alpha x_2^2 \\ \min & -5x_1 + 1.5x_2 + (\alpha+1)x_1^2 - (\alpha+1)x_1x_2 + (2-\alpha)x_2^2 \\ \min & -5x_1 + 1.5x_2 + (\alpha+1)x_1^2 - (3-\alpha)x_1x_2 + \alpha x_2^2 \\ \min & -5x_1 + 1.5x_2 + (\alpha+1)x_1^2 - (3-\alpha)x_1x_2 + (2-\alpha)x_2^2 \\ \min & -5x_1 + 1.5x_2 + (3-\alpha)x_1^2 - (\alpha+1)x_1x_2 + \alpha x_2^2 \\ \min & -5x_1 + 1.5x_2 + (3-\alpha)x_1^2 - (\alpha+1)x_1x_2 + (2-\alpha)x_2^2 \\ \min & -5x_1 + 1.5x_2 + (3-\alpha)x_1^2 - (3-\alpha)x_1x_2 + \alpha x_2^2 \\ \min & -5x_1 + 1.5x_2 + (3-\alpha)x_1^2 - (3-\alpha)x_1x_2 + (2-\alpha)x_2^2 \\ \text{s.t.} & x_1 + x_2 \geq 2 \\ & 2x_1 - x_2 \geq 4 \\ & x_1, x_2 \geq 0. \end{array} \quad (25)$$

Problem (25) is solved by weighting objectives, with $\alpha = 0.8$, *where an auxiliary problem like (21) can be obtained;*

$$\begin{array}{ll} \min & -5x_1 + 1.5x_2 + \omega 1(1.8x_1^2 - 1.8x_1x_2 + 0.8x_2^2) \\ & +\omega_2(1.8x_1^2 - 1.8x_1x_2 + 1.2x_2^2) + \omega_3(1.8x_1^2 - 2.2x_1x_2 + 0.8x_2^2) \\ & +\omega_4(1.8x_1^2 - 2.2x_1x_2 + 1.2x_2^2) + \omega_5(2.2x_1^2 - 1.8x_1x_2 + 0.8x_2^2) \\ & +\omega_6(2.2x_1^2 - 1.8x_1x_2 + 1.2x_2^2) + \omega_7(2.2x_1^2 - 2.2x_1x_2 + 0.8x_2^2) \\ & +\omega_8(2.2x_1^2 - 2.2x_1x_2 + 1.2x_2^2) \\ \text{s.t.} & x_1 + x_2 \geq 2, \; 2x_1 - x_2 \geq 4, \; x_1, x_2 \geq 0. \end{array} \quad (26)$$

Different optimal solutions to each of the weighting vectors is obtained by choosing the canonical base of $\mathbb{R}^8$ *as weighting vectors, as it is presented in Table 2.*

Table 2: Result of Problem 21 to each vector ω.

ω	x_1	x_2	FunObj
(1,0,0,0,0,0,0,0)	1.5114	0.4886	-3.8506
(0,1,0,0,0,0,0,0)	1.5521	0.4479	-3.7630
(0,0,1,0,0,0,0,0)	1.4688	0.5312	-4.1547
(0,0,0,1,0,0,0,0)	1.5096	0.4904	-4.0505
(0,0,0,0,1,0,0,0)	1.3854	0.6146	-3.0130
(0,0,0,0,0,1,0,0)	1.2705	0.3279	-2.9303
(0,0,0,0,0,0,1,0)	1.3558	0.6442	-3.3582
(0,0,0,0,0,0,0,1)	1.4018	0.5982	-3.2040

On the other hand, the approach described in 2.3 can be obtained by using the first and last weighting vector of Table 2. Hence, by taking $\omega_L = (1,0,0,0,0,0,0,0)$ *and solving (26) the optimal solution is* $x^* = [1.5114, 0.4886]$ *and value of objective function is -3.8506. Besides, by taking* $\omega_U = (0,0,0,0,0,0,0,1)$ *and solving (26) the optimal solution is* $x^* = [1.4018, 0.5982]$ *and value of objective function is -3.2040.*

Example 3 *Consider the following quadratic programming problem with all fuzzy coefficients:*

$$\begin{array}{ll} \min & (-6,-5,-4)x_1 + (1,1.5,2)x_2 + (1,2,3)x_1^2 + \\ & +(1,2,3)x_1x_2 + (0,1,2)x_2^2 \\ \text{s.t.} & x_1 + x_2 \geq 2 \\ & 2x_1 - x_2 \geq 4 \\ & x_1, x_2 \geq 0. \end{array} \quad (27)$$

The membership functions are

$$\mu_{-5}(c_1) = \begin{cases} c_1 + 6 & -6 \leq c_1 \leq -5 \\ -c_1 - 4 & -5 \leq c_1 \leq -4 \end{cases}$$

$$\mu_{1.5}(c_2) = \begin{cases} 2(c_2 - 1) & 1 \leq c_2 \leq 1.5 \\ 2(2 - c_2) & 1.5 \leq c_2 \leq 2 \end{cases}$$

$$\mu_2(c_1) = \begin{cases} c_1 - 1 & 1 \leq c_1 \leq 2 \\ 2 - c_1 & 2 \leq c_1 \leq 3 \end{cases}$$

$$\mu_1(c_2) = \begin{cases} c_2 & 0 \leq c_2 \leq 1 \\ 2 - c_2 & 1 \leq c_2 \leq 2 \end{cases}$$

From the membership function for these costs an interval

ISBN: 978-989-95079-6-8

representation according to (3) and (4) can be given:

$$c_1^{0.2} = [-5.8, -4.2], \quad c_1^{0.4} = [-5.6, -4.4],$$
$$c_1^{0.6} = [-5.4, -4.6], \quad c_1^{0.8} = [-5.2, -4.8]$$

$$c_2^{0.2} = [1.1, 1.9], \quad c_2^{0.4} = [1.2, 1.8],$$
$$c_2^{0.6} = [1.3, 1.7], \quad c_2^{0.8} = [1.4, 1.6]$$

$$c_3^{0.2} = [1.2, 2.8], \quad c_3^{0.4} = [1.4, 2.6],$$
$$c_3^{0.6} = [1.6, 2.4], \quad c_3^{0.8} = [1.8, 2.2]$$

$$c_4^{0.2} = [0.2, 1.8], \quad c_4^{0.4} = [0.4, 1.6],$$
$$c_4^{0.6} = [0.6, 1.4], \quad c_4^{0.8} = [0.8, 1.2]$$

with $c_1^0 = [-6, -4]$, $c_1^1 = [-5, -5]$, $c_2^0 = [1, 2]$, $c_2^1 = [1.5, 1.5]$, $c_3^0 = [1, 3]$, $c_1^3 = [2, 2]$ *and* $c_4^0 = [0, 2]$, $c_4^1 = [1, 1]$, *and now* $\mathbb{M} = \{0, 0.2, 0.4, 0.6, 0.8, 1\}$.

Different optimal solutions to each of the weighting vectors can be showed by the multi-objective method described choosing the canonical base of $\mathbb{R}^{32}$ *as weighting vectors. According to the previous examples, it can obtain the minimum and maximum values that form the optimal interval of this problem. The minimum value is -4.5021 where the optimal solution is* $[1.4792, 0.5208]$ *while the maximum value of this optimal interval is -2.6579 where the optimal solution is* $[1.1803, 0.2186]$.

4 Conclusions

This paper presented three approaches for solving fuzzy quadratic programming problems. A novel approach that transforms a quadratic programming problem with imprecise costs in the objective function into a parametric multiobjective quadratic programming problem. The other two approaches are methods published that are used in this work for comparison. Quadratic Programming is very important in theoretical and practical areas. When real-world applications are considered, vagueness appears in a natural way, and hence it makes perfect sense to think of fuzzy quadratic programming problems. In contrast to what happens in fuzzy linear programming problems, unfortunately until now not much research has been conducted to this important class of problems.

It was shown that the parametric multiobjective programming problem can obtain the optimal solutions of the other approaches because these solutions are a quadratic or linear combination of the optimal solutions obtained for each of weighting vectors. Hence, this shows that the parametric multiobjective approach contains the optimal solutions obtained by the others. Therefore, parametric approaches can be used as a general method to solve quadratic programming problems with uncertain costs in the objective function.

The authors aim to extend the line of investigation involving Fuzzy Quadratic Programming problems in order to try to solve practical real-life problems by facilitating the building of Decision Support Systems.

Acknowledgment

The authors want to thank the support provided by the Brazilian agency CAPES and the Spanish projects TIN2005-08404-C04-01 (70% of which are FEDER funds) and MINAS (TIC-00129-JA) for the support provided.

References

[1] E. Ammar and H. A. Khalifa. Fuzzy portfolio optimization a quadratic programming approach. *Chaos, Solitons and Fractals*, 18:1045–1054, 2003.

[2] S.S. Appadoo, S.K. Bhatt, and C.R. Bector. Application of possibility theory to investment decisions. *Fuzzy Optimization and Decision Making,*, 7:35–57, 2008.

[3] E. M. L. Beale. On quadratic programming. *Naval Research Logistics Quarterly*, 6:227–244, 1959.

[4] R. E. Bellman and L. A. Zadeh. Decision-marking in a fuzzy environment. *Management Science*, 17(4):B141–B164, 1970.

[5] G.R. Bitran. Linear multiple objective problems with interval coefficients. *Management Sciences*, 26(7):694–706, 1980.

[6] M. Delgado, J.L. Verdegay, and M.A. Vila. Imprecise costs in mathematical programming problems. *Control and Cybernetics*, 16(2):113–121, 1987.

[7] M. Delgado, J.L. Verdegay, and M.A. Vila. Relating different approaches to solve linear programming problems with imprecise costs. *Fuzzy Sets and Systems*, 37:33–42, 1990.

[8] C. A. Floudas, P.M. Pardalos, C. Adjiman, W. R. Esposito, Z. H. Gümüs, S. T. Harding, J. L. Klepeis, C. A. Meyer, and C. A. Sshweiger. *Handbook of test problems in local and global optimization*, volume 33 of *Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, Dordrecht, 1999.

[9] W. Hock and K. Schittkowski. *Test examples for nonlinear programming codes*, volume 187 of *Lecture Notes in Economics and Mathematical Systems*. Spring-Verlag, 1981.

[10] Y. J. Lai and C. L. Hwang. *Fuzzy mathematical programming : methods and applications*, volume 394 of *Lecture notes in economics and mathematical systems*. Springer, Berlin, 1992.

[11] S. T. Liu. Quadratic programming with fuzzy parameters: A membership function approach. Chaos, Solitons and Fractals. Article in press. doi:10.1016/j.chaos.2007.07.054, 2007.

[12] S. T. Liu. Solving quadratic programming with fuzzy parameters based on extension principle. In *FUZZ-IEEE 2007*, pages 1–5, London, UK, 2007. IEEE International Conference Fuzzy Systems.

[13] C. V. Negoita and D. A. Ralescu. *Applications of fuzzy sets to systems analysis*. Birkhauser Verlag, Stuttgard, 1975.

[14] D. Poole, A. Mackworth, and R. Goebel. *Computational intelligence: A logical approach*. Oxford University Press, 1998.

[15] S. J. Russel and P. Norvig. *Artificial intelligence: A modern approach*. Prentice Hall, Upper Saddle River, NJ, second edition, 2003.

[16] K. Schittkowski. *More test examples for nonlinear programming codes*, volume 282 of *Lecture Notes in Economics and Mathematical Systems*. Spring-Verlag, 1987.

[17] J. L. Verdegay, R. R. Yager, and P. P. Bonissone. On heuristics as a fundamental constituent of soft computing. *Fuzzy Sets and Systems*, 159(7):846–855, 2008.

[18] S. Wang and S. Zhu. On fuzzy portfolio selection problems. *Fuzzy Optimization and Decision Making,*, 1:361–377, 2002.

[19] P. Wolfe. The simplex method for quadratic programming. *Econometrica*, 27:382–398, 1959.

[20] L. A. Zadeh. A new direction in AI: Toward a computational theory of perceptions. *AI Magazine*, 22(1):73–84, 2001.