

A Modified Interactive Stability Algorithm for Solving Multi-Objective NLP Problems with Fuzzy Parameters in Its Objective Functions

Mohamed Abd El-Hady Kassem¹, Ahmad M. K. Tarabia², Noha Mohamed El-Badry^{2*}

¹Mathematics Department, Faculty of Science, Tanta University, Tanta, Egypt

²Mathematics Department, Faculty of Science, Damietta University, Damietta, Egypt

Email: mohd60_371@hotmail.com, a_tarabia@yahoo.com, *nooha_moh@yahoo.com

Received 17 August 2015; accepted 5 January 2016; published 11 January 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

This paper presents a modified method to solve multi-objective nonlinear programming problems with fuzzy parameters in its objective functions and these fuzzy parameters are characterized by fuzzy numbers. The modified method is based on normalized trade-off weights. The obtained stability set corresponding to α -Pareto optimal solution, using our method, is investigated. Moreover, an algorithm for obtaining any subset of the parametric space which has the same corresponding α -Pareto optimal solution is presented. Finally, a numerical example to illustrate our method is also given.

Keywords

Multi-Objective Nonlinear Programming, Stability, Trade-Off Method, Fuzzy Parameters

1. Introduction

Many real-life optimization problems have several conflicting objective functions that should be minimized or maximized simultaneously. Researchers and practitioners use various approaches to solve these multi-objective problems. Many authors such as Shih and Lee [1] proposed many approaches that integrated the simulation models with stochastic multiple objective optimization algorithms, many of which used the Pareto-based ap-

*Corresponding author.

How to cite this paper: El-Hady Kassem, M.A., Tarabia, A.M.K. and El-Badry, N.M. (2016) A Modified Interactive Stability Algorithm for Solving Multi-Objective NLP Problems with Fuzzy Parameters in Its Objective Functions. *American Journal of Operations Research*, 6, 8-16. <http://dx.doi.org/10.4236/ajor.2016.61002>

proaches that generated a finite set of compromise or tradeoff solutions.

In such multi-objective optimization problems, finding the best possible solution means trading off between the different objectives. Instead of a single optimal solution, we have a set of compromise solutions so-called Pareto optimal solutions where none of the objective function values can be improved without impairing at least one of the others. Indeed, in most multi-objective nonlinear programming problems, multi-objective functions usually conflict with each other, which means any improvement of one objective function can be achieved only at the expense of another.

Accordingly, our aim is to find the satisficing solution for the decision maker which is also Pareto optimal. However, formulating the multi-objective nonlinear programming problem which closely describes and represents the real decision situation reflects various factors of the real system. The description of the objective function and constraints involves many parameters whose possible values may be assigned by the experts.

Fuzzy nonlinear programming problem (FNLPP) is very useful in solving problems which are difficult to solve due to the imprecise, subjective nature of the problem formulation or have an accurate solution.

In an earlier work, Osman [2] introduced the notions of the stability set of the first kind and the second kind, and analyzed these concepts for parametric convex nonlinear programming problems. Osman and El-Banna [3] presented the qualitative analysis of the stability set of the first kind for fuzzy parametric multi-objective nonlinear programming problems. Kassem [4] dealt with the interactive stability of multi-objective nonlinear programming problems with fuzzy parameters in the constraints. Sakawa and Yano [5] introduced the concept of α -multi-objective nonlinear programming and α -Pareto optimality. Katagiri and Sakawa [6] dealt with fuzzy random programming, Loganathan and Sherali [7] presented an interactive cutting plane algorithm for determining a best-compromise solution to a multi-objective optimization problem in situations with an implicitly defined utility function. Jameel and Sadeghi [8] solved nonlinear programming problem in fuzzy environment. Recently, Elshafei [9] and Parag [10] gave an interactive stability compromise programming method for solving fuzzy multi-objective integer nonlinear programming problems.

Our motivation depends on developing the method given by Kassem [11] by adding normalized trade-off weights. The modified technique uses an interactive compromise programming algorithm to obtain the optimal stable solution of multi-objective nonlinear programming problems with fuzzy parameters in the objective function. In this algorithm a normalized trade-off weight for each objective function is considered. Moreover, our strategy is, if the decision maker (DM) is unsatisfied with the corresponding α -Pareto optimal solution with the degree α , the DM updates the degree α of α -level set by considering the stability set which has the same corresponding α -Pareto optimal solution.

In this paper, preliminary results are given in Section 2. Section 3 shows the stability set of the first kind. The problem formulation is given in Section 4. Section 5 deals an interactive stability compromise programming algorithm for solving multi-objective nonlinear programming problems with fuzzy parameters in the objective functions. Finally, an illustrative example is given to clarify the obtained results.

2. Preliminary Results

In this section several necessary basic concepts are recalled.

In general, the fuzzy multi-objective nonlinear programming problem (FMONLP) is represented as the following problem:

$$\begin{aligned} \text{(FMONLP)} \quad & \max (f_1(x, \tilde{a}_1), f_2(x, \tilde{a}_2), \dots, f_m(x, \tilde{a}_m)) \\ \text{subject to} \quad & X = \{x \in R^n \mid g_j(x) \leq 0, j = 1, \dots, k\}, x \in X, \end{aligned}$$

where $f_i(x, \tilde{a}_i)$ and $g_j(x)$ are continuously differentiable and concave functions for all $i = 1, 2, \dots, m$ and $j = 1, 2, \dots, k$. A nonempty set X is a convex and compact set, and $\tilde{a} = (\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_m)$ represents a vector of fuzzy parameters in the objective function $f_i(x, \tilde{a}_i), i = 1, 2, \dots, m$.

These fuzzy parameters are assumed to be characterized as the fuzzy numbers as given by Dubois and Prade [12]. It is appropriate to recall here that a real fuzzy number \tilde{a} is a convex continuous fuzzy subset of the real line whose membership function $\mu_{\tilde{a}}(a), a \in R$ and satisfies the following properties:

1. A continuous mapping from R to the closed interval $[0, 1]$.
2. $\mu_{\tilde{a}}(a) = 0$ for all $a \in (-\infty, a_1]$.
3. Strict increase on $[a_1, a_2]$.

4. $\mu_{\tilde{a}}(a) = 1$ for all $a \in [a_2, a_3]$.
5. Strict decrease on $[a_3, a_4]$.
6. $\mu_{\tilde{a}}(a) = 0$ for all $a \in [a_4, +\infty)$.

A possible shape of fuzzy number \tilde{p} is illustrated in **Figure 1**.

Definition 1. (Dubois and Prade [12]).

The α -level set of the fuzzy numbers \tilde{a}_i is defined as the ordinary set $L_\alpha(\tilde{a})$ for which degree of their membership function exceeds, the level α :

$$L_\alpha(\tilde{a}) = \{a \mid \mu_{\tilde{a}}(a) \geq \alpha, i = 1, \dots, m\}$$

For a certain degree of α , the problem FMONLP can be rewritten by using Sakawa and Yano's method [5]. In the following nonfuzzy α -multi-objective nonlinear programming problem form:

$$\begin{aligned} (\alpha\text{-MONLP}) \quad & \max (f_1(x, a_1), f_2(x, a_2), \dots, f_m(x, a_m)) \\ & \text{subject to } x \in X, a \in L_\alpha(\tilde{a}). \end{aligned}$$

Problem (α -MONLP) can be rewritten as the following form:

$$\begin{aligned} (\alpha\text{-MONLP}) \quad & \max (f_1(x, a_1), f_2(x, a_2), \dots, f_m(x, a_m)) \\ & \text{subject to } x \in X, A_i \leq a_i \leq B_i, i = 1, \dots, m, \end{aligned}$$

where A_i, B_i are lower and upper bounds on a_i for $i = 1, 2, \dots, m$.

Indeed, Sakawa, and Yano [5] consider that the membership function $\mu_{\tilde{a}}(a)$ is differentiable on $[a_1, a_4]$ and the problem FMONLP is stable, hence our new problem α -MONLP is also stable.

Definition 2. (α -Pareto optimal solution).

$x^* \in X$ is said to be an α -Pareto optimal solution to the (α -MONLP), if and only if there does not exist another $x \in X$, $a \in L_\alpha(\tilde{a})$. Such that $f_i(x, a_i) \geq f_i(x^*, a_i^*)$, $i = 1, 2, \dots, m$, with strictly inequality holding for at least one i , where the corresponding values of parameters a_i^* are called α -level optimal parameters.

For some (unknown) implicit utility function we have the following problem:

$$\begin{aligned} (\alpha M) \quad & \max U[f_1(x, a_1), f_2(x, a_2), \dots, f_m(x, a_m)] \\ & \text{subject to } x \in X, A_i \leq a_i \leq B_i, i = 1, \dots, m. \end{aligned}$$

where $U(\cdot)$ is a concave and continuously differentiable function. It is clear that (x^*, a^*) is an α -Pareto optimal solution of (α -MONLP) if and only if (x^*, a^*) is optimal solution of (αM).

3. Stability Set of the First Kind [11]

In this section we discuss the stability set of the first kind of the nonlinear programming problem with fuzzy parameters in the objective function.

Definition 3. (Stability set of the first kind).

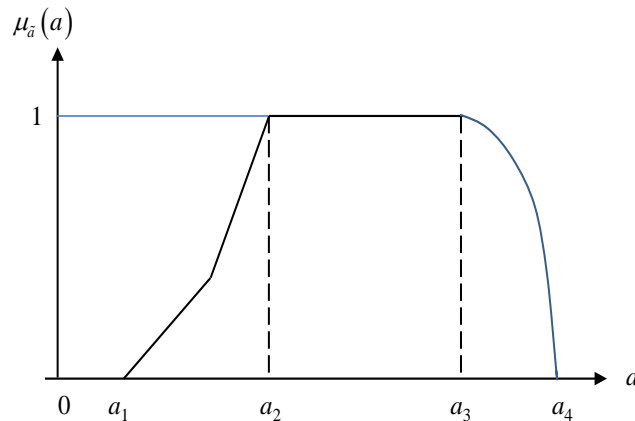


Figure 1. Membership function of fuzzy number \tilde{a} .

Suppose that $(\bar{A}, \bar{B}) \in R^{2m}$ with a corresponding α -Pareto optimal solution (\bar{x}, \bar{a}) , then the stability set of the first kind of (α -MONLP) corresponding to (\bar{x}, \bar{a}) , denote by $S(\bar{x}, \bar{a})$, is defined by

$$S(\bar{x}, \bar{a}) = \{(A, B) \in R^{2m} \mid (\bar{x}, \bar{a}) \text{ is an } \alpha\text{-Pareto optimal solution of } (\alpha\text{-MONLP})\}.$$

Let a certain (\bar{A}, \bar{B}) with a corresponding α -Pareto optimal solution (\bar{x}, \bar{a}) be given, then from the stability of problem (α -MONLP) there exist.

$(A, B) \in R^{2m}$, $v \in R^k$, λ, δ and $\mu \in R^m$ such that the following Kuhn-Tucker conditions are satisfied:

$$\sum_{i=1}^m \lambda_i \frac{\partial f_i}{\partial x_r}(\bar{x}, \bar{a}) + \sum_{j=1}^k v_j \frac{\partial g_j}{\partial x_r}(\bar{x}) = 0, \quad r = 1, \dots, n, \quad (1)$$

$$\lambda_i \frac{\partial f_i}{\partial \bar{a}_i}(\bar{x}, \bar{a}) - \delta_i + \mu_i = 0, \quad i = 1, \dots, m, \quad (2)$$

$$\sum_{i=1}^m \lambda_i = 1, \quad (3)$$

$$g_j(\bar{x}) \leq 0, \quad j = 1, \dots, k, \quad (4)$$

$$\bar{a}_i - B_i \leq 0, \quad i = 1, 2, \dots, m, \quad (5)$$

$$A_i - \bar{a}_i \leq 0, \quad i = 1, 2, \dots, m, \quad (6)$$

$$v_j g_j(\bar{x}) = 0, \quad j = 1, 2, \dots, k \quad (7)$$

$$\delta_i (\bar{a}_i - B_i) = 0, \quad i = 1, \dots, m, \quad (8)$$

$$\mu_i (A_i - \bar{a}_i) = 0, \quad i = 1, 2, \dots, m, \quad (9)$$

$$\lambda_i, \mu_i, \delta_i, v_j \geq 0, \quad i = 1, 2, \dots, m; j = 1, 2, \dots, k \quad (10)$$

The determination of the stability set of the first kind $S(\bar{x}, \bar{a})$ depends only whether any of the variables $\delta_i, i = 1, \dots, m$, and any of the variables $\mu_i, i = 1, 2, \dots, m$, given as Equation (2) and Equation (10), are positives or zero.

Given $\mu_i = 0, i \in I_1 \subseteq \{1, 2, \dots, m\}; \mu_i > 0, i \notin I_1$, as Equation (2) and Equation (10), then in order that the other Kuhn-Tucker conditions (6) and (9) yield

$$A_i \leq \bar{a}_i, \quad i \in I_1; A_i = \bar{a}_i, \quad i \notin I_1.$$

Given $\delta_i = 0, i \in I_2 \subseteq \{1, 2, \dots, m\}; \delta_i > 0, i \notin I_2$, as Equation (2) and Equation (10), then in order that the other Kuhn-Tucker conditions (5) and (8) yield

$$B_i \geq \bar{a}_i, \quad i \in I_2; B_i = \bar{a}_i, \quad i \notin I_2.$$

Let

$$\begin{aligned} S_{I_1, I_2}(\bar{x}, \bar{a}) &= \{(A, B) \in R^{2m} \mid A_i \leq \bar{a}_i, i \in I_1; A_i = \bar{a}_i, i \notin I_1 \\ &\quad B_i \geq \bar{a}_i, i \in I_2; B_i = \bar{a}_i, i \notin I_2\} \\ S(\bar{x}, \bar{a}) &= \bigcup_{I_1, I_2} S_{I_1, I_2}(\bar{x}, \bar{a}) \end{aligned}$$

4. Problem Formulation

A modified interactive stability method is the interactive nonlinear programming with fuzzy parameters in the objective functions to obtain the solution of (FMONLP) problem. In this method the DM asks to specify the degree α of the α -level set. For the DM's degree α , the corresponding optimal solution is given by solving the fol-

lowing problem:

$$\max_{x \in X} \min_{1 \leq i \leq m} \sum_{i=1}^m \tilde{w}_{ij}^l (f_i(x, a_i) - \bar{f}_i),$$

subject to $x \in X, a \in L_\alpha(\tilde{a}), i = 1, \dots, m,$

or equivalently

$$(\alpha W^l) \quad \max \eta$$

subject to $\eta \leq \sum_{i=1}^m \tilde{w}_{ij}^l (z_i - z_i^h), h = 0, 1, 2, \dots, l,$

$$(x, a) \in X \times L_\alpha(\tilde{a}),$$

where $\eta \approx \min_{1 \leq i \leq m} \sum_{i=1}^m \tilde{w}_{ij}^l (f_i(x, a_i) - \bar{f}_i)$, and \tilde{w}_{ij}^l normalized trade-off weights for $i \leq j$

$$\tilde{w}_{ij}^l = \frac{\left(\frac{\partial U(z)}{\partial z_j} \right) / \left(\frac{\partial U(z)}{\partial z_i} \right)}{\sum_{j=1}^k \left(\frac{\partial U(z)}{\partial z_j} \right) / \left(\frac{\partial U(z)}{\partial z_i} \right)}, \quad i \leq j,$$

where $z = (z_1, z_2, \dots, z_m), z_i \equiv f_i(x, a_i)$ for $i = 1, 2, \dots, m$, and $z^l \equiv (f_1(x^l, a_1^l), \dots, f_m(x^l, a_m^l))$.

The following lemma establishes an important characteristic of the problem αP^l .

Lemma1.

Let $(\bar{x}, \bar{a}, \bar{\eta})$ be a solution for the problem αW^l , and let $\bar{z} = (f_1(\bar{x}, \bar{a}_1), \dots, f_m(\bar{x}, \bar{a}_m))$. Then (\bar{x}, \bar{a}) is an α -Pareto optimal solution to the α -MONLP problem.

Proof:

Suppose that (\bar{x}, \bar{a}) is not α -Pareto optimal solution, then there exists $(x^*, a^*) \in X \times L_\alpha(\tilde{a})$ such that $z^* > \bar{z}$ and $z^* \neq \bar{z}$, where $z^* \equiv (f_1(x^*, a_1^*), \dots, f_m(x^*, a_m^*))$, then

$$\bar{\eta} = \min_{h=0,1,\dots,l} \sum_{i=1}^m \tilde{w}_{ij}^l (\bar{z} - z^h) < \min_{h=0,1,\dots,l} \sum_{i=1}^m \tilde{w}_{ij}^l (z^* - z^h),$$

which contradicts with the optimality of $(\bar{x}, \bar{a}, \bar{\eta})$. This completes the proof.

5. Interactive Algorithm

Following the above discussion, the interactive algorithm to derive the satisficing solution for the DM from among the α -Pareto optimal solution set is constructed. The steps are marked with numbers involving interaction with the DM.

Step 1: Ask the DM to select the initial values of $\alpha, (0 \leq \alpha \leq 1)$.

Step 2: Determine the α -level set of the fuzzy numbers.

Step 3: Convert the FMONLP in the form of α -MONLP and select an initial feasible point (x^l, a^l) , set $l = 0$ and let $z^l = (f_1(x^l, a_1^l), \dots, f_m(x^l, a_m^l))$.

Step 4: The DM specify values of \tilde{w}_{ij}^l at (x^l, a^l) for $i = 1, 2, \dots, m$.

Step 5: Solve problem αW^l . Suppose $(x^{l+1}, a^{l+1}, \eta^{l+1})$ be an optimal solution, and let $z^{l+1} = (f_1(x^{l+1}, a_1^{l+1}), \dots, f_m(x^{l+1}, a_m^{l+1}))$. Set $l = l + 1$.

Step 6: If $\eta^l = 0$, go to Step 7. Otherwise go to Step 4.

Step 7: Determine the stability set of the first kind $S(\bar{x}, \bar{a})$.

Step 8: If the DM is satisfied with the current values of the objective functions and α of α -Pareto solution, go to Step 9. Otherwise, ask the DM to update the degree α and return to Step 2.

Step 9: Terminate with (x^l, a^l) as the final solution.

6. Numerical Illustration Example

In this section, we give an example to illustrate the theoretical part have mention above. Our computation is carried out by utilizing MAPLE 13 (see [Appendix](#)).

Consider the following fuzzy multi-objective nonlinear programming problem

$$\begin{aligned} \text{(FMONLP)} \quad & \max (x_1 + \tilde{a}_1, x_2 + \tilde{a}_2) \\ \text{subject to} \quad & x \in X = \{(x_1, x_2) \mid -x_1 + x_2 \leq 3, x_1^2 + x_2^2 \leq 25, x_1, x_2 \geq 0\}, \end{aligned}$$

with

$$\mu_{\tilde{a}_i}(a_i) = \begin{cases} 0 & -\infty < a_i \leq c_1 \\ \frac{a_i - c_1}{c_2 - c_1} & c_1 < a_i < c_2 \\ 1 & c_2 \leq a_i \leq c_3 \\ \frac{a_i - c_4}{c_3 - c_4} & c_3 < a_i < c_4 \\ 0 & c_4 \leq a_i < \infty \end{cases}$$

where $I = 1, 2$ and the values of $c_j, (j = 1, 2, 3, 4)$ are given as in the following table:

\tilde{a}_i	c_1	c_2	c_3	c_4
\tilde{a}_1	3.8	4	4.8	5
\tilde{a}_2	1	2	3	4

Let $z_1 = f_1(x, a_1) = x_1 + a_1$ and $z_2 = f_2(x, a_2) = x_2 + a_2$. The DM's utility function U is assumed to be as the following:

$$U = -(z_1 - 20)^2 - 2(z_2 - 10)^2$$

Step 1: Suppose that the DM select $\alpha = 0.9$.

Step 2: $L_{0.9}(\tilde{a}) = \{(a_1, a_2) \mid 3.98 \leq a_1 \leq 4.82 \text{ and } 1.9 \leq a_2 \leq 3.1\}$

Step 3: The α -MONLP problem becomes:

$$\begin{aligned} & \max (x_1 + a_1, x_2 + a_2) \\ \text{subject to} \quad & x \in X, a \in L_{0.9}(\tilde{a}) \end{aligned}$$

Let us select the feasible point $(x^0, a^0) = (3.18, 2.9, 4.82, 3.1)$ as the starting solution. Hence, the corresponding point in the objective space is: $z^0 \equiv (8, 6)$.

Step 4: The normalized trade off weights w^0 at (x^0, a^0) is: $w^0 = (0.6, 0.4)$.

Step 5: Solve the following problem:

$$\begin{aligned} & (\alpha W^0) \max \eta \\ \text{subject to} \quad & \eta \leq 0.6x_1 + 0.4x_2 + 0.6a_1 + 0.4a_2 - 7.2 \\ & x \in X, a \in L_{0.9}(\tilde{a}) \end{aligned}$$

The outputs of our calculation are show in [Table 1](#).

Step 6: Since $\eta^l \neq 0$, then return to Step 4.

Repeat Steps 4 - 6 till $\eta^l \approx 0$, otherwise, go to Step 7.

Step 7: The corresponding stability set of the first kind is:

$$S(x^9, a^9) = \{(\bar{A}, \bar{B}) \mid \bar{A}_1 \leq 4.82, \bar{B}_1 = 4.82, \bar{A}_2 \leq 3.1, \bar{B}_2 = 3.1\}.$$

Step 8: Suppose that the DM is satisfied with the z^9 and $\alpha = 0.9$. Go to Step 9.

Table 1. The solution of the problem (αW^l) , $l = 1, 2, \dots, 9$.

Iteration l	(x^l, a^l)	η^l	z^l
1	(4.16025, 2.7735, 4.82, 3.1)	0.53755	(8.98025, 5.8735)
2	(4.003358, 2.9955173, 4.82, 3.1)	0.00528124	(8.823358, 6.0955173)
3	(4.1004068, 2.8612347, 4.82, 3.1)	0.0019716	(8.9204068, 5.9612347)
4	(4.038235, 2.948331, 4.82, 3.1)	0.0008195	(8.858235, 6.048331)
5	(4.078066, 2.892987, 4.82, 3.1)	0.00033366	(8.898066, 5.992987)
6	(4.0497106, 2.93254906, 4.82, 3.1)	0.0001699	(8.8697106, 6.03254906)
7	(4.0724333, 2.9009114, 4.82, 3.1)	0.000108795	(8.8924333, 6.0009114)
8	(4.0554198, 2.9246487, 4.82, 3.1)	0.0000611	(8.8754198, 6.0246487)
9	(4.0665451, 2.9091599, 4.82, 3.1)	≈ 0	(8.8865451, 6.0091599)

Step 9: The final solution is $(x^9, a^9) = (4.0665451, 2.9091599, 4.82, 3.1)$.

7. Conclusions

In this paper we have integrated different methods like normalized trade-off in [9] and compromised method by Kassem [11] to find a modified method which can be used to solve and obtain an optimal solution to the assumed problem.

The modified technique is based on the reformulation of the given problem and in such way that enables us to solve it easily. A new form of the assumed problem is obtained; hence using a computer program the stability of its solution is studied. Moreover, depending on stability calculation, the validity of the optimal solution has been ensured.

References

- [1] Shih, C.J. and Lee, H.W. (2004) Level-Cut Approaches of First and Second Kind for Unique Solution Design in Fuzzy Engineering Optimization Problems. *Tamkang Journal of Science and Engineering*, **7**, 189-198.
- [2] Osman, M. (1977) Qualitative Analysis of Basic Notions in Parametric Convex Programming I (Parameters in the Constraints). *Aplikace Matematiky*, **22**, 318-332.
- [3] Osman, M. and El-Banna, A. (1993) Stability of Multiobjective Nonlinear Programming Problems with Fuzzy Parameters. *Mathematics and Computers in Simulation*, **35**, 321-326. [http://dx.doi.org/10.1016/0378-4754\(93\)90062-Y](http://dx.doi.org/10.1016/0378-4754(93)90062-Y)
- [4] Kassem, M. (1995) Interactive Stability of Multiobjective Nonlinear Programming Problems with Fuzzy Parameters in the Constraints. *Fuzzy Sets and Systems*, **73**, 235-243. [http://dx.doi.org/10.1016/0165-0114\(94\)00317-Z](http://dx.doi.org/10.1016/0165-0114(94)00317-Z)
- [5] Sakawa, M. and Yano, H. (1989) Interactive Decision Making for Multiobjective Nonlinear Programming Problems with Fuzzy Parameters. *Fuzzy Sets and Systems*, **29**, 315-326. [http://dx.doi.org/10.1016/0165-0114\(89\)90043-2](http://dx.doi.org/10.1016/0165-0114(89)90043-2)
- [6] Katagiri, H. and Sakawa, M. (2011) Interactive Multiobjective Fuzzy Random Programming through the Level Set-Based Probability Model. *Information Sciences*, **181**, 1641-1650. <http://dx.doi.org/10.1016/j.ins.2011.01.003>
- [7] Loganathan, G.V. and Sherali, H.D. (1987) A Convergent Interactive Cutting-Plane Algorithm for Multiobjective Optimization. *Operations Research*, **35**, 365-377.
- [8] Jameel, A. and Sadeghi, A. (2012) Solving Nonlinear Programming Problem in Fuzzy Enlivenment. *International Journal of Contemporary Mathematical Sciences*, **7**, 159-170.
- [9] Elshafei, M.M. (2006) Interactive Stability of Multiobjective Integer Nonlinear Programming Problems. *Applied Mathematics and Computation*, **176**, 230-236.
- [10] Pendharkar, P.C. (2013) Scatter Search Based Interactive Multi-Criteria Optimization of Fuzzy Objectives for Coal Production Planning. *Engineering Applications of Artificial Intelligence*, **26**, 1503-1511. <http://dx.doi.org/10.1016/j.engappai.2013.01.001>
- [11] Kassem, M. (2007) Interactive Stability Cutting-Plane Algorithm for Multiobjective Nonlinear Programming Problems. *Applied Mathematics and Computation*, **192**, 446-456. <http://dx.doi.org/10.1016/j.amc.2007.03.037>
- [12] Dubois, D. and Prade, H. (1980) Fuzzy Sets and Systems: Theory and Application. Academic Press, New York.

Appendix

A maple program for solving multi-objective nonlinear programming (MONLP) problems and stability of this solution.

>restart;

With (Optimization):

>N1:= solve ({(a[1] - 3.8)/(4 - 3.8)>= 0.9}, [a[1]]);

>N2:= solve ({(a[1] - 5)/(4.8 - 5)>= 0.9}, [a[1]]);

N3:= solve ({(a[2] - 1)/(2 - 1)>= 0.9}, [a[2]]);

>N4:= solve ({(a[2] - 4)/(3 - 4)>= 0.9}, [a[2]]);

>U:= - (z[1] - 20) ^ 2 - 2 * (z[2] - 10) ^ 2;

r[1]:= diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 3.18 + 4.82, z[2] = 2.9 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 3.18 + 4.82, z[2] = 2.9 + 3.1}, r[2]);

Q1:= Maximize (e, {e <= 0.6 * a[1] + 0.6 * x[1] + 0.4 * a[2] + 0.4 * x[2] - 7.2, -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

r[1]:= diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 4.16025 + 4.82, z[2] = 2.7735 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 4.16025 + 4.82, z[2] = 2.7735 + 3.1}, r[2]);

Q2:= Maximize(e, {e <= 0.572 * x[1] + 0.428 * x[2] + 0.572 * a[1] + 0.428 * a[2] - ((0.572 * 8.98025) + (0.428 * 5.8735)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

r[1]:= diff (U, z[1])/(diff(U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff(U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 4.003358 + 4.82, z[2] = 2.9955173 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 4.003358 + 4.82, z[2] = 2.9955173 + 3.1}, r[2]);

Q3:= Maximize (e, {e <= 0.589 * x[1] + 0.411 * x[2] + 0.589 * a[1] + 0.411 * a[2] - ((0.589 * 8.823358) + (0.411 * 6.0955173)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

r[1]:= diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 4.1004068 + 4.82, z[2] = 2.8612347 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 4.1004068 + 4.82, z[2] = 2.8612347 + 3.1}, r[2]);

Q4:= Maximize (e, {e <= 0.578 * x[1] + 0.422 * x[2] + 0.578 * a[1] + 0.422 * a[2] - ((0.578 * 8.9204068) + (0.422 * 5.9612347)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

r[1]:= diff (U, z[1])/(diff(U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff(U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 4.038235 + 4.82, z[2] = 2.948331 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 4.038235 + 4.82, z[2] = 2.948331 + 3.1}, r[2]);

Q5:= Maximize (e, {e <= 0.585 * x[1] + 0.415 * x[2] + 0.585 * a[1] + 0.415 * a[2] - ((0.585 * 8.858235) + (0.415 * 6.048331)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

>r[1]:= diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));

r[1]:= subs ({z[1] = 4.078066 + 4.82, z[2] = 2.892987 + 3.1}, r[1]);

r[2]:= subs ({z[1] = 4.078066 + 4.82, z[2] = 2.892987 + 3.1}, r[2]);

Q6:= Maximize (e, {e <= 0.58 * x[1] + 0.42 * x[2] + 0.58 * a[1] + 0.42 * a[2] - ((0.58 * 8.898066) + (0.42 * 5.992987)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume = nonnegative);

>r[1]:= diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));

r[2]:= diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));


```

r[1]: = subs ({z[1] = 4.0497106 + 4.82, z[2] = 2.93254906 + 3.1}, r[1]);
r[2]: = subs ({z[1] = 4.0497106 + 4.82, z[2] = 2.93254906 + 3.1}, r[2]);
Q7: = Maximize (e, {e <= 0.584 * x[1] + 0.416 * x[2] + 0.584 * a[1] + 0.416 * a[2] - ((0.584 * 8.8697106) + (0.416 *
6.03254906)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, as-
sume = nonnegative);
>r[1]: = diff (U, z[1])/(diff (U, z[1]) + diff (U, z[2]));
r[2]: = diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));
r[1]: = subs ({z[1] = 4.0724333 + 4.82, z[2] = 2.9009114 + 3.1}, r[1]);
r[2]: = subs ({z[1] = 4.0724333 + 4.82, z[2] = 2.9009114 + 3.1}, r[2]);
Q8: = Maximize (e, {e <= 0.581 * x[1] + 0.419 * x[2] + 0.581 * a[1] + 0.419 * a[2] - ((0.581 * 8.8924333) + (0.419 *
6.0009114)), -x[1] + x[2] <= 3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume
= nonnegative);
>r[1]: = diff(U, z[1])/(diff (U, z[1]) + diff (U, z[2]));
r[2]: = diff (U, z[2])/(diff (U, z[1]) + diff (U, z[2]));
r[1]: = subs ({z[1] = 4.0554198 + 4.82, z[2] = 2.9246487 + 3.1}, r[1]);
r[2]: = subs ({z[1] = 4.0554198 + 4.82, z[2] = 2.9246487 + 3.1}, r[2]);
Q9: = Maximize (e, {e <= 0.583 * x[1] + 0.417 * x[2] + 0.583 * a[1] + 0.417 * a[2] - ((0.583 * 8.8754198) + (0.417 *
6.0246487)), -x[1] + x[2] <=3, x[1] ^ 2 + x[2] ^ 2 <= 25, 3.98 <= a[1], a[1] <= 4.82, 1.9 <= a[2], a[2] <= 3.1}, assume
= nonnegative);

```