Design and Implementation of MOLP Problems with Fuzzy Objective Functions Using Approximation and Equivalence Approach

Stephen Gbenga Fashoto *Member, IAENG,* Sulaiman Sani *Member, IAENG*

***Abstract*—Of late, pressure and need to manage emergencies in production systems with multiple objective functions are mounting. Information and Communication Technology (ICT) is one system where optimal response is needed for high level management amidst complex conflicting selection criteria. This paper designs an approach for solving this class of complexity in Multi-Objective Linear Programming problems (MOLP) with fuzzy objective functions. We take advantage of the Embedding theorem for fuzzy numbers to measure complex functions with several quantities within a production channel. Using C++ (Netbeans) with MATLAB, some design comparison and analysis with Nearest Interval Approximation (NIA) based defuzzification works are carried out. The gains of the design presented here are demonstrated in some benchmark problems solved for ease of application.**

***Index Terms*—MOLP, fuzzy numbers, defuzzification, embedding theorem.**

I. Introduction

Mathematical programming problems with several objective functions are features of naturally occurring processes of science, engineering and technology; Andreas and Wu-Sheng [2], Bunn [11], Charnes and Cooper [3], Charnes and Cooper [4]. In telecommunications engineering for instance, traffic processes are known to exhibit several functions within a transportation channel; Levoy et al. [29]. Again, traffic control robots must deal with complex functions for higher order optimal control; Drenick and Ko [40]. In multiprocessing computing, multi-linear programming is a pre-requisite for high level operational computing; Drenick and Drenick [37]. Essentials of network calculus require the specific treatment of several competing goals in programming; Georgiev et al. [16] for complex problems such as the ones exemplified above; Oberman and Ruan [31]. Such goals cannot arbitrarily be squeezed within the narrow framework of a single objective function without maximizing the risk of invalidation. Instances and examples in line with the endorsed paradox of Ignazio [25] and the Arrow impossibility theorem of Zeleny [34] are clear in Ergott [32], Drenick [40], Eiselt et al. [17], Goicoechea et al. [5] and Drenick and Drenick [37]. Beside the reasons above, it is known in optimization parlance that conflicting goals onto a multi-objective programming framework involve parameter uncertainty capable of stunting critical choices that render results less desirable. As a result, new methodologies are needed for reasons to do with optimal decision programming. For more on the benefit of MOLP problems on several production systems, see Oberman and Ruan [31], Simon [20], Gizegorreswski [38]. A feasible challenge in programming a complex production system is seeking a definite distribution for randomized statements, quantities and variables that are somewhat infinite in structure. The case of MOLP network problem with open boundaries suffices; Warid et al. [42], Abdulqadeer et al. [7], Li et al. [23], Xu and Jin [10].

Here, decision makers face the task of approximating functions when seeking preferred solution. Zhang and Li [39] observed that most MOLP problems with open boundaries have infinite structures. The case of MOLP problem of an open road linking two points in space suffices. If the surface is long enough, the process of execution is time consuming principally due to dimension issues considered for effective programming. With more care, such problems are daunting if not impossible to complete. The difficulty here is in finding suitable distribution without loss of reality. Another challenge facing real life MOLP problems is their nonlinearity; Das and Dennis [21] principally due to the magnitude of embedded fuzzy quantities; Sarmad et al. [1]. Here, the case of the internet traffic programming suffices; Boxma and Cohen [36] and Medhi [24]. Thus, it is in place to construct new methods for MOLP problems; Rakowska et al. [26], Katopis and Lin [15] and Eschenauer et al. [18] for effective operations.

Our aim is to design one approach for somewhat more harder class of MOLP problems with quantifiers like most, many, few, not very many, almost all, frequently and so on. These quantifiers are found in traffic channels of telecommunication and computer systems where complex fuzzy quantities such as " most passenger alight", " almost all passenger alight", " few passenger alight", must be considered for better management given “optimum optimorun”. These day to day quantifiers of ICT are infinite in magnitude as well as nonlinear in dimension. A MOLP problem involving such quantifiers must take account of conflicting criteria without loss of reality for optimal solution.

The case of MOLP for betting games suffices in this respect; Jiang et al. [19]. Thus, a resort to sufficing solution along Zadeh’s extension principle and Simon’s bounded rationality principle is imperative and well posed; see Luhandjula and Rangoaga [33], Franck et al. [29], Georgiev et al. [16]. Within this resort, we design a novel defuzzification procedure for such quantities passing through a communication channel with infinite functions and derived results are implemented on $C++$ with NetBeans and MATLAB.

Manuscript received April, 2020; revised December, 2020.

Stephen Gbenga Fashoto is a Post-Doctoral Fellow in the Department of Decision Sciences, University of South Africa, Pretoria-South Africa (e-mail: sgfashoto@uniswa.sz).

Sulaiman Sani is a Senior Lecturer in the Faculty of Science and Engineering, Department of Mathematics, University of Eswatini, Kwaluseni-Eswatini (e-mail: ssulaiman@uniswa.sz).

Furthermore, we provide examples demonstrating how such quantifiers can be defuzzified under optimum optimorun for effective or efficient programming of MOLP problems.

II. Related Literature

Given a classical universal set X, a real valued function $\mu_A : X \to [0,1]$ is called the membership function of A and defines the fuzzy set A of X; Luhandjula and Rangoaga [33]. The ordered pairs $(x, \mu_A(x))$ for all $x \in X$ is a measurable space. The above definition covers the case even where X is of infinite cardinality; Zhu et al. [28]. The set of support of A is the set of all elements $x \in X$ for which the pair $(x, \mu_A(x)) \in A$ for all $\mu_A(x) > 0$. If A is fuzzy with finite support $\{a_1, a_2, a_3, ..., a_m\}$, then A is described as follows; $A = \{\frac{\mu_1}{a_1} + \frac{\mu_2}{a_2} + \frac{\mu_3}{a_3} + ... + \frac{\mu_m}{a_m}\}$ where each $\mu_i = \mu_A(a_i)$ for $i = 1, ..., m$; Nguyen and Nguyen [43].

In what follows, $A \subseteq X$ is denoted by $\breve{a}$ with membership function μ. It is worthy of note that μ has an inverse on $\Re$ or any of its subset by the completeness axiom. Thus, μ has a support defined as the crisp set $Supp(\mu) = \overline{\{x \in \Re \mid \mu(x) > 0\}}$. Again, the core of μ denoted by $\ker(\mu) = \{x \in \Re \mid \mu(x) = 1\}$. If $\ker(\mu) \neq \emptyset$, then such a μ-membership is normal in $A \subseteq X$; Saade [22]. Suppose $\alpha \in \Re$ exists such that $\mu^\alpha = \{x \in \Re \mid \mu(x) \geq \alpha\}$. Then μ^α is the α-cut of μ; Quevedo [27] and is convex on X if it is at least quasi-concave.

Bana and Coroianu [6] have shown that every convex differentiable fuzzy functions with support $Supp(\mu)$ is isomorphic to some convex programming problem under certain conditions. The isomorphism in this case is induced naturally by the Euclidean metric known for the open interval $\Re$.

Thus, given a fuzzy set consisting of convex and differentiable functions $f, g_1, ..., g_m : \Re^n \to \Re$, the support $\bar{x} \in X$ naturally solves a convex programming problem of the form $\min f(x) : g_i(x) \leq h_i(x) \quad i \in \{1, 2, ..., m\}$ for some $h_i \in \Re$ provided that there exists ζ_i such that (1.) $\nabla f(\bar{x}) + \sum_{i=1}^{m} \zeta_i \nabla g(\bar{x}) = 0$. (2.) $g(\bar{x}) - h_i \leq 0$. (3.) $\zeta_i \geq 0$. (4.) $\zeta_i(h_i - g(\bar{x})) = 0$. Consequently, constraining a given MOLP problem lies in its uniqueness given embedded constraints.

Again, Landoli et al. [30] and Bana and Coroianu [6] have proved that, given a fuzzy space $\mathbf{F}(\Re)$, a fuzzy set $A \in \mathbf{F}(\Re)$ on $[a, b] \subseteq \Re$ has a preserved ambiguity provided that

$$a \geq \int_0^1 \left(\alpha + \frac{1}{2}\right) A_L d\alpha + \int_0^1 \left(-\alpha + \frac{1}{2}\right) A_U d\alpha \qquad (1)$$

and

$$b \leq \int_0^1 \left(-\alpha + \frac{1}{2}\right) A_L d\alpha + \int_0^1 \left(\alpha + \frac{1}{2}\right) A_U d\alpha \qquad (2)$$

In this case, the interval $[a, b] \subseteq \Re$ of $A \in \mathbf{F}(\Re)$ is unique for a cut α of A. We make the following proposition for those α cuts in the neighbourhood of complex fuzzy constraints subject in this work.

Proposition 1: Suppose $f, g_1, ..., g_m : \Re^n \to \Re$ are convex and differentiable functions with a cut α. If $\mathbf{F}(\Re)$ is bounded given the ambiguity of fuzzy set $A \in \mathbf{F}(\Re)$, then there exists a closed interval $I = [a, b] \subseteq \Re$ such that arbitrary $g_i's$ resides in I.

Proof: We proof for two functions $f, g \in \mathbf{F}(\Re)$. By Stone-Weistrass theorem, the measure p such that

$$p(f, g) = \sup\{d(f(x), g(x)) \mid x \in X\} \qquad (3)$$

is finite and $d(f(x), g(x)) = p(f, g)$ is continuous since both f, g are differentiable. Applying the triangular inequality on the right side of (3) yields

$$d(f(x), g(x)) \leq d(f(x), g(y)) + \xi + d(f(y), g(x)) \qquad (4)$$

Hence,

$$d(f(x), g(x)) - \xi \leq d(f(x), g(y)) + d(f(y), g(x)) \qquad (5)$$

By symmetry,[1]

$$\xi - d(f(x), g(x)) \leq d(f(x), g(y)) + d(f(y), g(x)) \qquad (6)$$

So that

$$\mid h(x) - h(y) \mid \leq \mid d(f(x), g(x)) - d(f(y), g(y)) \mid \qquad (7)$$

$$\leq d(f(x), g(y)) + d(f(y), g(x)) \geq 0 \qquad (8)$$

satisfying proposition (3). Thus, there exists $\bar{x} \in \Re$ which solves a differentiable convex programming problem. To show that arbitrary $f \in \mathbf{F}(\Re)$ has a limit point $x \in \Re$, observe that f and g are continuous since they are differentiable. Consequently, there exists $\epsilon > 0$ and $\delta > 0$ such that

$$\mid h(x) - h(y) \mid \leq \mid d(f(x), f(y)) + d(g(x), g(y)) \mid < \epsilon \qquad (9)$$

whenever $d(x, y)) < \delta$. Hence arbitrary function $h \in \mathbf{F}(\Re)$ is continuous. By Bolzano-Weistrass boundedness theorem, $I \in \Re$ exists a.s. It remains to be shown that $p(f, g)$ is a metric on $\mathbf{C}(\Re)$. By Stone-Weistrass, the measure $p(f, g)$ is finite. Let $f, g_1, g_2 \in \mathbf{F}(\Re)$, we show that

$$p(f, g_2) \leq p(f, g_1) + p(g_1, g_1) \qquad (10)$$

by (3) and the arguments of (4) through (8) triangular inequality holds. ■

III. Methods and Assumptions

We consider an optimisation problem consisting of several fuzzy objective functions $f, g_1, ..., g_m : \Re^n \to \Re$ under crisp constraints satisfying (1) and (2) respectively. We supposed that $f_{i's}$ are differentiable functions in $\Re$ satisfying proposition 1. Without loss of generality, we restrict the $f_{i's}$ to be semi-deterministic with the objective that

$$P1 = \left\{\max(\bar{f_1}(x), \bar{f_2}(x), \bar{f_3}(x), ..., \bar{f_m}(x))\right\}, \quad x \in X \qquad (11)$$

Subject to

$$a_1X_1 + a_2X_2 ... + a_nX_n \leq h_1 \qquad (12)$$

$$b_1X_1 + b_2X_2 ... + b_nX_n \leq h_2 \qquad (13)$$

$$c_1X_1 + c_2X_2 ... + c_nX_n \leq h_3 \qquad (14)$$

$$... \qquad (15)$$

$$... \qquad (16)$$

$$X_1 \geq 0, X_2 \geq 0, ..., X_n \geq 0 \qquad (17)$$

where $\bar{f_i}(x), i = 1, 2, \ldots\ldots, m$ are fuzzy functions and $a_{i's}, b_{i's}, c_{i's}, ..., h_{i's}$ are vector valued functions.

[1] $\xi = d(f(y), g(y))$.

The problem presented in (11) through (17) depicts real-life problems to do with optimal allocation management where day to day activities are governed by continuity of functions with ambiguities. Fuzzy objective functions (FoF) for this class of problems provide systematic framework for dealing with uncertain quantities like most, many, few, not very many, almost all, commonly known for these problems.

We supposed that $\bar{f}_i(x)), i = 1, 2, \ldots\ldots, m$ are semi-linear to guarantee existence of solution. There are two approaches of interest for the MOLP problem in question. The Approximation approach (NIA) and the Equivalence approach (EA). As stated in section II, these approaches exist in the literature only that to the best of our knowledge, the EA is not applied on the specific fuzzy quantities of interest in this work.

The NIA approach here is closely related to that of Luhandjula and Rangoaga [33] directly. Here, it is coded and presented for comparison purpose principally. The idea is to approximate $\bar{f}_i(x), i = 1, \ldots, m$ in (11) by the NIA and then solve an optimization problem. Afterwards, the same $\bar{f}_i(x), i = 1, ..., m$ is solved using the EA over a discrete interval; see Appendix A and Appendix B. These Appendices represent pseudo codes designed primarily for (11) and implemented using $C++$ with NetBeans IDE 7.0.1 and MATLAB R2013a. The hardware requirements are on personal computer equipped with intel (R) $Core^{TM}2$ Duo CPU with a processor speed of 2.5GHz, 3GB of RAM, 32bit Operating System, 135GB of Hard Disk and runs under windows Vista.

IV. Numerical Results

Appendix A and Appendix B present pseudo codes for approaching (11) under the NIA and the EA. From Appendix A, we observed that the NIA approach for (11) has few problems to do with the execution of complex ambiguities (11) is subjected to; that is (12) to (17). The designed pseudo codes for $(P1)^{'''}$ of $P1$ via the NIA approach finds it easier to produce gains sequel to coarseness and computational inefficiency. We understand that the ease of coarseness emanates from the treatment of the entire FoF over the continuous interval for executing $P1$.

Again, we observed that overlapping during execution is minimal and eventually, made the process less harder to execute. This way, the coupled effect of FoF over the continuous fuzzy interval creates less time for the NIA approach to execute $(P1)^{'''}$ for $P1$ leading to good computational efficiency stated above. Moreover, the entire execution process is completed over the fuzzy space, suggesting that the NIA pseudo codes designed here is effective in generating $(P1)^{'''}$ for $P1$ over the interval. Again, the NIA pseudo codes treats the entire process in few steps by locating the maximum FoF at random and carrying out the process much more faster than the EA; see Table IV.

On the other hand, the EA (Appendix B) seeking Pareto optimal solution of $P1$ as a discretized problem $(PI)^v$ of $P1$ strikes a balance between computational tractability and effectiveness. This development is traced to the discretized intervals required by the EA approach. Clearly, it becomes tough for the EA to execute $(PI)^v$ of $P1$ over equivalent intervals as in proposition 1 without overlapping. This overlapping in execution makes the EA execute the task less efficiently compared to the NIA approach. The EA must identify the maximum FoF in each sub interval, finds the supremum FoF before execution. The magnitude of the pseudo codes in this case speaks volumes in this respect; see Appendix B.

We concluded that a Pareto optimal solution of $(PI)^v$ is tantamount to that of $(PI)^{'''}$ where $g_{i's}$ of proposition 1 are replaced by $Kg_{i's}$ in the case of the EA. This way, when h is small as in proposition 1, $Kg_{i's}$ approximate $g_{i's}$ for this class of MOLP problems effectively. Moreover, the greater the value of K, the less efficient is the EA and vice versa. This claim holds good in view of the increasing size of overlaps in finding the supremum FoF. Fortunately, the EA transcends over all FoF in the execution process before an integration process.

This aspect places more effectiveness points in favour of the approach compared to the NIA approach that only scans over an interval. Thus, The EA is effective with high computation tractability in terms of the size of a problem while the NIA approach is efficient with slow computation in terms of memory usage and execution time. The high effectiveness of the EA; see Table V especially as the number of FoF increases becomes weaker when the average FoF values is small. Finally, we concluded that the two approaches studied in this work are able to successfully convert the FoF to deterministic ones with the NIA approach making gains in efficiency and the EA in effectiveness.

Theorem 2: An execution point $x^* \in [a, b]$ is Pareto optimally efficient for $P1$ if the overlap constant $K \overrightarrow{1}$ from above.

Proof: This is in view of the effect of K as in Appendix B. Additionally, it is sequel to the relation $Kg_{i's} \sim g_{i's}$ when h of proposition 1 is small. Now, suppose that $x^* \in [a, b]$ is Pareto optimal for $P1$. Then there exists no other point $x \in [a, b]$ of proposition 1 such that

$$\bar{f}_1(x^*) \leq \bar{f}_i(x); \quad \forall\ i \in [1, 2, 3, ..., m] \tag{18}$$

Again, there is no $x \in [a, b]$ such that

$$\bar{f}_i(x^*) \leq \bar{f}_i(x); \quad \forall\ i \in [1, 2, 3, ..., m] \tag{19}$$

By (1) and (2), $\bar{f}_i(.)$ is order preserving for each $x \in [a, b]$ including x^*. Thus

$$\Pi \bar{f}_i(x^*) \leq \Pi \bar{f}_i(x) = K \Pi \bar{f}_i(x); \quad \forall i \in [1, 2, 3, ..., m] \tag{20}$$

The LHS of (20) are the $g_{i's}$ and the RHS are the $Kg_{i's}$ which holds only if $K = 1$ so that $Kg_{i's}$ of the EA goes to $g_{i's}$ of the NIA approach. Given that the NIA approach is efficient, it follows directly that the EA is efficient at $K = 1$ and completes the proof. ■

Theorem 3: The bridge between the EA and the NIA approaches is the scaling constant $K \in \Re$.

Proof: By Theorem 2, each $g_{i's}$ of the NIA has an equivalent $Kg_{i's}$ in the EA. Suppose by contradiction that x^* is Pareto optimal for $(P1)^v$ and not Pareto optimal for $(P1)^{iv}$. This implies that Theorem 2 does not hold at $K = 1$. Again, there is $x^* \in [a, b]$ such that

$$Kg_{i\alpha}(x^*) \leq Kg_{i\alpha}(x^*),\ \forall (i, \alpha) \in [a, b] \tag{21}$$

And no $K \in \Re$ such that

$$Kg_{i\tilde{\alpha}}(x^*) \leq Kg_{i\tilde{\alpha}}(x^*),\ \forall (i, \tilde{\alpha}) \in [a, b] \tag{22}$$

Thus, by (22), x^* is not an optimal point for $(P1)^{iv}$. Consequently, there exists $x \in [a, b]$ such that

$$g_{i\alpha}(x^*) \leq g_{i\alpha}(x),\ \forall (i, \alpha) \in [a, b] \tag{23}$$

And

$$g_{1\alpha' s}(x^*) \leq g_{1\alpha' s}(x),\ \forall (1, \alpha) \in [a, b] \tag{24}$$

Let $\alpha \in [0, 1]$ be an arbitrary point such that for some $\omega_.$, we have $0 \leq \omega_j(\alpha) \leq \delta_{ij}$. Then by (23) and (24), we have

$$\sum_{j=0}^{m} \omega_j(\alpha)[g_{i\alpha j}(x^*) - g_{i\alpha j}(x)] \leq 0,\ \forall\ j \leq J \tag{25}$$

Since α is arbitrarily chosen, then (25) implies directly that there is such $x \in [a, b]$ such that (22) holds good for some $K \in \Re$. This contracts our claim in (21) and so if x^* is Pareto optimal for $(P1)^v$, it follows that it is Pareto optimal for $(P1)^{iv}$ given $K \in \Re$. Hence K is the bridge in view of the continuity assumption[2] on $g's$ as in Lemma 1.

■

V. Main Results

Theorem 4: Let F be a choice correspondence defined on some Ω^n satisfying $(PI)^v$ and $(PI)^{'''}$ for PI. Then for any $x, y^m \in \Re^+$ such that $y^m \to y \neq x$, there exists $M \neq K = 1$ such that

$$\{y\} = F(ch\{x, y\}) \Rightarrow F(ch(\{x, y^m\}))] \tag{26}$$

And

$$\{x\} = F(ch\{x, y\}) \Rightarrow F(ch(\{x, y^m\}))] \tag{27}$$

Proof: Take $\epsilon > 0$ such that x is an exterior point of $N_\epsilon(x)$. By the NIA, there exists a $\delta > 0$ such that for some $K \in \Re^+$, we have $F(K) \subseteq N_\epsilon(y)\ \forall\ K \in N_\delta(ch\{x, y\})$. Since $\lim_{m\to\infty} y^m = y$, it implies directly that $\lim_{m\to\infty}(ch\{x, y^m\}) = ch(\{x, y\})$. Consequently, there exists $M > 0$ such that $ch\{x, y^m\} \in N_\delta ch(\{x, y\})$ for all $m > M$. Hence, we are guaranteed that whenever $m > M$, we have $F(ch\{x, y^m\}) \in N_\epsilon(y)$. Finally, since x is an exterior point of $N_\epsilon(y)$ then (26) holds good. Similar arguments holds good for (27) trivially in respect of the EA.

■

Theorem 5: Let F be a choice correspondence defined on some Ω^n satisfying $(PI)^v$ and $(PI)^{'''}$ for PI. Then, there does not exist such F on Ω^n that satisfies both $(P1)^v$ and $(P1)^{'''}$ for computational efficiency and effectiveness.

Proof: Suppose by contradiction that such F correspondence for computational efficiency and effectiveness on Ω^n for the NIA and the EA exists. Let $x, y \in \Re^+$ such that for some $i, j \in \{1, ..., n\}$,we have $x_i > y_i$ and $x_j < y_j$ in either efficiency or effectiveness. By the EA, there exists a $\lambda^* > \lambda_* > 1$ such that $F(ch\{x, \lambda y\}) = x$ (theorem 4) for all $\lambda \in (1, \lambda_*)$ and $F(ch\{x, \lambda y\}) = \lambda y$ (part 2 of theorem 4) for all $\lambda \in (\lambda^*, \infty)$. Thus, $\lambda \mapsto F(ch\{x, \lambda y\})$ is not continuous on (λ, λ^*) for either the computational efficiency or effectiveness respective of the NIA and and the EA. This discontinuity suffices for the nonexistence of a unique NIA and the EA schemes for both computational efficiency and effectiveness. ■

[2]ET: Execution Time, AA: Approximation Approach.

Theorem 6: There is a Choquet bargaining solution between the $(P1)^{'''}$ of the NIA and the $(P1)^v$ of the EA relative to computational efficiency and effectiveness.

Proof: We show that there exists a monotonic capacity υ on $\{1, 2, 3, ..., n\}$ such that any choice F of computational efficiency over effectiveness (or vice versa) on a set $S \in \Omega^n$ is

$$F(S) = \arg\max_{x\in S} \int x d\upsilon\ \ \forall\ S \in \Omega^n \tag{28}$$

Assume by contradiction that F satisfies both the NIA and the EA on computational efficiency and effectiveness without a Choquet bargain. Then by theorem 5, there exists a continuous and strictly monotonic map $W : \Re^n \to \Re$ such that

$$F(S) = \arg\max_{x\in S} W(S)\ \ \forall\ S \in \Omega^n \tag{29}$$

(29) implies that given W and for any $x, y \in \Re^n$, we have $W(x + y) = W(x) + W(y)$; hence comonotonic. Since W is continuous and monotonic, there exists some $\xi_x, \xi_y > 1$ such that $W(x) = W(\xi_x I_n)$ and also that $W(y) = (\xi_y I_n)$ implying that $\{x, \xi_x I_n\} = F(ch\{x, \xi_x I_n\})$ and $\{y, \xi_y I_n\} = F(ch\{y, \xi_y I_n\})$ (theorems 4 and 5). But since W is comonotonic with x and $\xi_x I_n$, we have $F(\{x + y, \xi_x I_n + y\}) = \{x + y, \xi_x I_n + y\}$ by the NIA and the EA implying that $W(x + y) = W(\xi_x I_n + y)$. The same analysis holds for y. Since W is strictly additive, by Schmeidler representation theorem, W has a monotonic capacity υ such that $\upsilon(A) = W(\chi_A)\ \ \forall\ A \subseteq \{1, 2, ..., n\}$ and the theorem holds goods.

■

VI. Discussion and Conclusion

This work implements the MOLP Problems with Fuzzy Objective Functions using the NIA and the EA. The aim is to design an approach for somewhat more harder class of MOLP problems with quantifiers like most, many, few, not very many, almost all, frequently and so on. We proved analytic results vital for smoothing the two approaches studied in this work. Numerical results are demonstrated and pseudo codes are designed with the proposed pseudo-codes statistically evaluated using IBM SPSS statistics 20.

It is clearly shown in Appendix C that a trade-off relationship exists between ET and memory usage in the EA and the NIA. The Pearson correlation coefficient for this relationship is 0.896 showing a strong relationship between ET and memory usage statistically significant at $p = 0.016 (p < 0.05$ for a two tailed test).

On the other hand, Appendix D shows that there is a weak trade-off relationship between ET and magnitude of FoF in the EA and the NIA as established by the Pearson correlation coefficient (0.449) which shows that they are statistically less significant at $p = 0.372 (p > 0.05$ for a two tailed test).

Finally, Appendix E shows that there is a moderate trade-off relationship between memory usage and magnitude of FoF in the EA and the NIA as established by the Pearson correlation coefficient (0.651) at significant level $p = 0.651 (p > 0.05$for a two tailed test).

Generally, efficiency and effectiveness are key criteria for evaluating performance of a MOLP problem with FoF;

Mangaraj [9] and Mouzas [41]. The effectiveness of approach given increasing magnitude of fuzzy objective values is of special interest in this study. On the other hand, there is the subject of efficiency of an algorithm principally based on execution time and/or memory usage. Widely, algorithms with fast execution time and sizeable memory usage are more efficient than their converse cases; Folorunso et al. [35]. Thus, pseudo-codes designed for optimization purpose must achieve either the effectiveness superiority or the efficiency one. On some specific notes, there are interests in sacrificing more execution time and memory space algorithms and vice versa for high level effectiveness; Lofti et al. [14], Nicoara [13].

In this work, we follow the effectiveness in design paradigm in defining the optimal solution from the two pseudo-codes implemented on the NIA and the embedding theorem leading to the Pareto approximation (EA). Table V and Table VI have shown that the execution time (ET) and memory usage for both algorithms are good in the general sense. Though, the NIA values are better compared with those of the EA sequel to the later using more information involving defuzzification of the complex fuzzy quantities; (see Tables I, II and III) more than the NIA approach. More explicitly, the EA has to create an integral space for the union of all discrete spaces over the process leading to loss of memory usage and time for higher effectiveness and reliability so to speak; see Table V and Table VI. By, so doing, it ensures the consideration of all ambiguities attached to given FoF in the solution process of the MOLP problem strongly more than the NIA.

One concludes that the EA on open MOLP problems like the case studied here; (that is $P1$) with complex FoF is more effective; Golany [8] for smart systems where effectiveness is special in the application of total ambiguities for optimal control.

On the basis of quality of solution, it can also be inferred from Tables IV, V, VI that the superiority of the EA over the NIA approach is out of question and superb. This is sequel to the distinctive treatment of total ambiguities because of the discretization employed in this work. Our result outperforms the approximations via NIA of Xu and Jin [10] in effectiveness and reliability making the design presented in this work fit for smart systems of transportation and communication working under complex interactions.

In conclusion, from the statistical perspective, Appendices C, D and E have shown that there is less relationship between effectiveness (magnitude of the FoF) and efficiency (ET and memory usage) in the two approaches implying that the value of K might not be unity in the real sense.

Acknowledgement

The authors are grateful to all the literature sources used, the anonymous reviewers and editors of this paper.

VII. Appendices

A. The NIA Design for PI

```
INPUT: e,f,alpha_upper,alpha_lower,lamda1,
lamda2,g11,g12,g21,g22,n
OUTPUT: mew1,mew2, x1, x2
PROCESS
funct evaluate_integrad(e, f, alpha_upper,
alpha_lower)
begin
return ( ((f - e) * pow(alpha_upper, 2)
/2 + e * alpha_upper) - ((f - e) *
pow(alpha_lower, 2) / 2 + e * alpha_
lower));
end
funct c(n,e, f)
begin
return evaluate_integrad(e, f, 1, 0);
end

funct mew_1( c,i_size_of_array, lamda1,
lamda2,g11, g12,mew2, result)
begin
if (g11 == 0) return false;
result = -1*(lamda1 * c[1] + lamda1
*c[2] + lamda2 * c[5] + lamda2 * c[6]
+ mew2 * g12) / g11;
return true;
end

funct mew_2(c,i_size_of_array, lamda1,
lamda2,g11,g12, g21,g22,result)
begin
if ((g11 * g22 - g12 * g21) == 0)
return false;
result = -1*((lamda1 * c[3] * g11
+ lamda1* c[4] * g11 +
lamda2 * c[7] * g11 + lamda2
*c[8] * g11)- (lamda1 * c[1]
* g21 + lamda1 * c[2] * g21
+ lamda2 * c[5]
* g21 + lamda2 * c[6] * g21))
/ (g11 * g22 - g12 * g21);
return true;
end
main function
int main(int argc, char**argv)
begin
if (load_C_from_file("input.txt", C,
i_size_of_array,lamda1, lamda2,
g11, g12, g21, g22) == NULL) return 0;
c = new float[i_size_of_array + 1];
for (int i = 1; i <= i_size_of_array;
 i++)
begin
split_text_into_two_numbers(C_
global[i], e, f);
c[i] = evaluate_integrad(e, f,
1, 0);
cout << "INTEGRAND(" << e << ","
<< f << ") or
c[" << i << "]: " << c[i] << "\n";
end
if (load_lambdas_from_file("lambdas.
txt", s_lamdas,
i_count_of_lambdas)
== NULL) return 0;
s_lamdas = C_global;
```

```
if( load_coefficients_from_file
("coefficient.txt",
s_coefficient, i_size_of_array)
== NULL) return 0 ;
s_coefficient = C_global;
i_count_of_lambdas = 20 ;
for (int i = 1; i <= i_count_of_
lambdas; i++)
begin
lamda1 = rand();
lamda2= rand() ;
if (mew_2(c, i_size_of_array,
lamda1,lamda2, g11, g12, g21,
g22, mew2) ==
false) return 0;
if (mew_1(c, i_size_of_array,
lamda1,lamda2, g11, g12, mew2,
mew1) == false)
return 0;
if (mew1>0 && mew2>0)
begin
cout << "(" << lamda1 <<
"," << lamda2
<< ","
<< mew1 << "," << mew2
<<")\n" ;
end
end
split_text_into_six_numbers
(s_coefficient[1], a1,
b1, c1, d1, e1, f1);
x_1(  a1,    b1,   c1,   d1,
e1,   f1,   x1) ;
x_2(  a1,    b1,   c1,   d1,
e1,   f1,   x2) ;
cout << "The Satisficing Solution
="<<"(" << x1<<","<<x2<<")\n";
return 0;
end
```

B. The EA Design for $P1$

```
Step 1    Fix an acceptable upper bound
for the roughness of the grid h,
say epsilon >0.
Step 2  Read data of  [( P1)]^’’’.
Step 3  Put i=1,l_i=2, alpha_(l_i1 )=0,
alpha_(l_i2 )=1.
S_li={alpha_(l_i1 ),alpha
 _(l_i1 )}
Step 4    Compute
h=[max]_(alpha in I) [min]
_(1 leq j leq s) | alpha-
alpha_(l_ij ) |,
where s is such that
alpha_(l_is )=1.
Step 5    check whether h leq epsilon.
If yes, go to Step 6
otherwise
take a finer discretization
 of I,
 S_(l_(i+1) )
 Put l_i=l_(i+1) and go to Step 4.
Step 6   Write [( P1)]^v.
Step 7 Find a Pareto optimal solution of
[( P1)]^v.
Step 8 Print x^* is a satisficing solution
of [( P1)]^’’’.
Step 9    Stop.
```

```
The Pseudo codes:

INPUT: alpha, lambda
OUTPUT: m_c1,m_c2
PROCESS
Funct  getFirstVector(alpha , lamda)
begin
return new ListItem(lamda * (1 + alpha)
    / 2, lamda * (9 + alpha) / 5);
end

funct  getSecondVector(alpha, lamda)
begin
return new ListItem(lamda * (3 - alpha)
     / 2, lamda * (3 - alpha));
end

funct  getThirdVector(alpha ,lamda)
begin
return new ListItem(lamda * 2 * alpha,
     lamda * alpha);
end
funct  getFourthVector(alpha , lamda)
begin
return new ListItem( lamda * (3 - alpha),
     lamda * (3 - 2 * alpha));
end

main function
int main(int argc, char** argv)
begin
if (load_coefficients_from_file
("coefficients1.txt",
s_coefficient,i_size_of_array)
 == NULL) return 0;
s_coefficient = C_global;
ListItem *pRootOfList =
new ListItem(0, 0);
ListItem *pListItem;
cout << "The Objective Functions of
the Discretized
Problem" << "\n";float alpha = 0;
int n = 20;
for (int m = 1; m <= n; m++)
begin
if (load_lambdas_from_file("lambda.txt",
s_lamdas, i_count_of_lambdas)
== NULL) return 0;
s_lamdas = C_global;
alpha = 0;
iLammda = 1;
```

```
while (alpha <= 1)
begin
pListItem = getFirstVector1(alpha,
atof(s_lamdas[iLammda++].c_str()));
if (pListItem->setNext(pRootOfList))
pRootOfList = pListItem;
pListItem->print();
pListItem = getSecondVector(alpha,
atof(s_lamdas[iLammda++].c_str()));
pListItem->setNext(pRootOfList);
if (pListItem->setNext(pRootOfList))
pRootOfList = pListItem;
pListItem->print();
pListItem = getThirdVector(alpha,
atof(s_lamdas[iLammda++].c_str()));
pListItem->setNext(pRootOfList);
if (pListItem->setNext(pRootOfList))
pRootOfList = pListItem;
pListItem->print();
pListItem = getFourthVector(alpha,
atof(s_lamdas[iLammda++].c_str()));
pListItem->setNext(pRootOfList);
if (pListItem->setNext(pRootOfList))
pRootOfList = pListItem;
pListItem->print();
alpha = alpha + 0.25;
cout << "\n";
end
cout << "The Single objective
Optimization Solution=" <<
pRootOfList->sumItemsOnList
(pRootOfList)->m_c1() << "x1" << "+";
cout << pRootOfList->sumItemsOnList
(pRootOfList)->m_c2() << "x2 \n";
end
return 0;
end
```

C. Correlations on Execution time and Memory Usage

```
              execution_time Memory_usage
 Pearson Correlation   1         .896*
execution_time
    Sig. (2-tailed)            .016
         N             6           6
Pearson Correlation
Memory_usage           .896*       1
Sig. (2-tailed)       .016
         N             6           6
*Correlation is significant at the 0.05
level (2-tailed).
```

D. Correlations on Execution time and Magnitude of FoF

```
    execution_time Magnitude of FoF
 Pearson Correlation   1         .449
execution_time
    Sig. (2-tailed)            .0372
         N             6          6
Pearson Correlation   .449         1
Magnitude of FoF
Sig. (2-tailed)      .372
         N             6           6
*Correlation is significant at the 0.05
level (2-tailed).
```

E. Correlations on Magnitude of FoF and Memory use

```
    execution_time Magnitude of FoF
 Pearson Correlation   1         .651
Magnitude of FoF
    Sig. (2-tailed)            .161
         N              6          6
Pearson Correlation   .651        1
Memory use
Sig. (2-tailed)      .161
         N             6           6
*Correlation is significant at the 0.05
level (2-tailed).
```

F. Numerical Demonstrations

We demonstrate these variations by constructing certain fuzzy MOLP problems $E1$, $E2$, $E3$ and provide numerical illustrations[3] for added clarity. Here,

$$E1: \quad \max(\tilde{c}_1^1 x_1 + \tilde{c}_2^1 x_2, \tilde{c}_1^2 x_1 + \tilde{c}_2^2 x_2) \tag{30}$$

Subject to

$$X_1 + X_2 \leq 6 \tag{31}$$

$$2X_1 + X_2 \leq 9 \tag{32}$$

$$X_1 \geq 0, X_2 \geq 0 \tag{33}$$

The triangular FoF of $E1$, $E2$ and $E3$ are respectively given in Tables I, II, III for some constraints as given below

TABLE I
Triangular fuzzy numbers for E1

$\tilde{c}_1^1$	$\tilde{c}_2^1$	$\tilde{c}_1^2$	$\tilde{c}_2^2$
$(0.5, 1, 1.5)$	$(1.8, 2, 3)$	$(0, 2, 3)$	$(0, 1, 3)$

$$E2: \quad \max(\tilde{c}_1^1 x_1 + \tilde{c}_2^1 x_2 + \tilde{c}_3^1 x_3, \varpi, \tilde{c}_1^3 x_1 + \tilde{c}_2^3 x_2 + \chi) \tag{34}$$

Subject[4] to

$$2X_1 + X_2 + X_3 \leq 2 \tag{35}$$

$$X_1 - X_2 - X_3 \leq -1 \tag{36}$$

$$X_1 + 3X_2 + X_3 \leq 5 \tag{37}$$

$$X_1 \geq 0, X_2 \geq 0, X_3 \geq 0 \tag{38}$$

$$E3: \quad \max(\tilde{c}_1^1 x_1 + \tilde{c}_2^1 x_2 + \tilde{c}_3^1 x_3 + \tilde{c}_4^1 x_4, ., ., \varsigma \tag{39}$$

Subject to

$$3X_1 + 2X_2 + 5X_3 + X_4 \leq 55 \tag{40}$$

$$X_1 + 2X_2 + X_3 + X_4 \leq 26 \tag{41}$$

$$X_1 + X_2 + 3X_3 + 2X_4 \leq 30 \tag{42}$$

$$2X_1 + X_2 + 3X_3 + X_4 \leq 10 \tag{43}$$

$$X_1 \geq 0, X_2 \geq 0, X_3 \geq 0, X_4 \geq 0 \tag{44}$$

[3] $\varsigma = \tilde{c}_1^4 x_1 + \tilde{c}_2^4 x_2 + \tilde{c}_3^4 x_3 + \tilde{c}_4^4 x_4)$.
[4] $\varpi = \tilde{c}_1^2 x_1 + \tilde{c}_2^2 x_2 + \tilde{c}_3^2 x_3$, $\chi = \tilde{c}_3^3 x_3$.

TABLE II
Triangular fuzzy numbers for E2

$\tilde{c}_1^1$	$\tilde{c}_2^1$	$\tilde{c}_3^1$	$\tilde{c}_1^2$	$\tilde{c}_2^2$
$3, 4, 5$	$6, 7, 8$	$9, 210, 11$	$12, 13, 14$	$15, 16, 17$
$\tilde{c}_2^3$	$\tilde{c}_3^1$	$\Delta = \tilde{c}_3^2$	$\Delta = \tilde{c}_3^3$	
$18, 19, 20$	$21, 22, 23$	$24, 25, 26$	$27, 28, 29$	

TABLE III
Triangular fuzzy numbers for E3

$\tilde{c}_1^1$	$\tilde{c}_2^1$	$\tilde{c}_3^1$	$\tilde{c}_4^1$	$\tilde{c}_1^2$
$1, 2, 3$	$4, 5, 6$	$7, 8, 9$	$10, 11, 12$	$13, 14, 15$
$\tilde{c}_2^2$	$\tilde{c}_3^2$	$\Delta = \tilde{c}_4^2$	$\Delta = \tilde{c}_1^3$	$\Delta = \tilde{c}_2^3$
$16, 17, 18$	$19, 20, 21$	$22, 23, 24$	$25, 26, 27$	$28, 29, 30$
$\tilde{c}_3^3$	$\tilde{c}_4^3$	$\Delta = \tilde{c}_1^4$	$\Delta = \tilde{c}_2^4$	$\Delta = \tilde{c}_3^4$
$31, 32, 33$	$34, 35, 36$	$37, 38, 39$	$40, 41, 42$	$43, 44, 45$
$\tilde{c}_4^4$				
$46, 47, 48$				

TABLE IV
Execution time Analysis

Constraints	ET using EA(ms)	ET using AA (ms)
2	266	47
3	896	63
4	1000	281

TABLE V
Memory Usage Analysis

FoF Size	Approach	Memory Usage (Kbyte)
2	EA	15.7
2	NIA	15.3
3	EA	16.8
3	NIA	16.1
4	EA	17.4
4	NIA	16.3

TABLE VI
Solution for E

Aproach	Solution	$\overline{FoF}$
EA	$(0, 6)$	6.14
NIA	$(3, 3)$	4.01
EA	$(0, 0, 2)$	36.17
NIA	$(0.3, 1.7, -0.3)$	23.14
EA	$(0, 11.4, 0, 3.6)$	353.07
NIA	$(9.21, 1.32, 0, 4.47)$	340.46

REFERENCES

[1] A. A. Sarmad, F. A. Jameel and A. Saaban, ”Homotopy Perturbation Method Approximate Analytical Solution of Fuzzy Partial Differential Equation” *IAENG International Journal of Applied Mathematics*, vol. 49, no 1, pp. 22-28, 2019.

[2] A. Andreas and L. Wu-Sheng, *“Practical Optimization Algorithm and Engineering Applications*,” Springer, New York, 2007.

[3] A. Charnes and W. W. Cooper, “Chance-Constrained Programming”, *Management Sciences*, vol. 6, no. 1, pp. 73-79, 1959.

[4] A. Charnes and W. W. Cooper, *“Management Models and Industrial Applications of Linear Programming”*, John Wiley and Sons Inc., New York, 1961.

[5] A. Goicoechea, D. R. Hansen and L. Duckstein, *“Multiobjective Decision Analysis with Engineering and Business Applications”*, John Wiley and Sons Inc., New York, 1982.

[6] A. I. Bana and L. Coroianu, ”Nearest interval, triangular and trapezoidal approximation of a fuzzy number preserving ambiguity”, *International Journal of Approximate Reasoning,* vol. 53, pp. 805–836, 2012.

[7] A. O. Hamadameen and, Z. M. Zainuddin, ”Multiobjective Fuzzy Stochastic Linear Programming Problems in the 21st Century”, *UTM. Life Science Journal*, vol. 10, no. 4, pp. 1-32, 2013.

[8] B. Golany, ”An Interactive MOLP Procedure for the Extension of DEA to effectiveness analysis”, *Journal of Operational Research Society*, vol. 39, no. 8, pp. 725-734, 1988.

[9] B. K. Mangaraj, ”Relative effectiveness analysis under fuzziness”, *Procedia Computer Science*, vol. 102, pp. 231-238, 2016.

[10] B. Xu and Y. J. Jin, ”Multiobjective dynamic topologyoptimization of truss with interval parameters based on interval possibility degree”, *Journal of vibrations and control*, vol. 20, no. 1, 66-81, 2012.

[11] D. W. Bunn, *“Applied Decision Analysis”,* Mcgraw Hill, New York, 1984.

[12] E. A. Oke and L. Zhou, ”Revealed group preferences on non-convex choice problems”, *Economic Theory*, vol. 13, pp. 671-687, 1999.

[13] E. S. Nicoara, ”Performance Measures for multi-objective optimization algorithms”, *Seria matematica-infomatica*, vol. LIX, pp. 19-28, 2007.

[14] F. H. Lofti, A. A. Noora, G. R. Jahanshahloo, M. Khodabakhshi and A. Payan, ”A linear programming approach to test efficiency in multi-objective linear fractional programming problems”, *Applied Mathematical Modelling*, vol. 34, no. 12, pp. 4179-4183, 2010.

[15] G. A. Katopis and J. G. Lin, ”Non inferiority of Controls under Double Performance Objectives Minimal Time and Minimal Energy”, *Proceedings of the Hawaii Int. Conf. Syst. Sci.*, Honolulu, Hawaii, pp. 129-131, 1974.

[16] G. Georgiev, I. Balabanova, S. Kostadinova and R. Dimova, *IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Structure synthesis of ANFIS classifier for teletraffic system resources identification,* 6-9 June 2016.

[17] H. A. Eiselt, G. Pederzoli and C. L. Sandblom, *“Continuous Optimization Models,”* Walter deGruyter and Company, Berlin, 1987.

[18] H. Eschenauer, J. Koski and A. Osyczka, ”Multicriteria Design Optimization”, *Berlin Springer Verlag* publication, 1990.

[19] H. Jiang, D. Qiu and Y. Xing, ”Solving Multi-objective Fuzzy Matrix Games via Fuzzy Relation Approach”, *IAENG International Journal of Applied Mathematics*, vol. 49, no. 3, pp. 339-343, 2019.

[20] H. Simon, *“A Behavioral Model of Rational Choice in Models of Man, Social and Rational,”* Macmillan, new York, 1957.

[21] I. Das and J. E. Dennis, ”Normal-Boundary Intersection: A New Method for Generating the Pareto Surface in Nonlinear Multicriteria Optimization Problems”, *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631-657, 1998.

[22] J. J. Saade, ”Mapping convex and normal fuzzy sets”, *Fuzzy Sets and Systems,* vol. 81, pp. 25l 256, 1996.

[23] J. L. Jiuping and X. M. Genb, ”A class of multiobjective linear programming model with fuzzy random coefficients”, *Mathematical and Computer Modelling,* vol. 44, no. 11–12, pp. 1097-1113.

[24] J. Medhi, Stochastic Models in Queueing theory. 2nd Edition, Academic Press, California, USA, 2003.

[25] J. P. Ignizio, *“Linear Programming in Single- and Multiple-objective Systems,”* Prentice-Hall Inc., Engelwood Cliffs, 1982.

[26] J. Rakowska, R. T. Haftka and L. T. Watson, ”Tracing the Efficient Curve for Multi Objective Control Structure Optimization”, *Computing Systems in Engineering,* vol. 2, no. 6, pp. 461-471.

[27] J. R. N. Quevedo, ”Fuzzy sets. A way to represent ambiguity and subjetivity”, *Boleton de Matemcaticas,* vol. 24, no. 1, pp. 57-88, 2017.

[28] K. Zhu, J. Wang and Y. Yang, ”A Study on Z-Soft Fuzzy Rough Sets in BCI-Algebras”, *IAENG International Journal of Applied Mathematics,* vol. 50, no. 3, pp. 577-583, 2020.

[29] L. Franck, J. Edward, B. Anthony, O. Monforta, N. Robinc, P. Bretel, ”Formation and migration of transverse bars along a tidal sandy coastdeduced from multi-temporal Lidar datasets”, *Marine Geology*, vol. 342, pp. 39–52, 2013.

[30] L. Landoli, E. Marchione, C. Ponsiglione and G. Zollo, ”Computing ambiguity in complex systems with Fuzzy logic”, *economic review,* vol. XIV(1), 2009.

[31] M. A. Oberman and Y. Ruan, *”An Efficient Linear Programming Method for Optiml Transportation”*, 2000.

[32] M. Ehrgott, ”A Discussion of Scalarization Techniques for Multiobjective Integer Programming”, *Annals of Operations Research,* vol. 147, no. 1, pp. 343-360, 2006.

[33] M. K. Luhandjula and M. J. Rangoaga, ”An approach for solving a Multiobjective Programming model with fuzzy objective functions”, *European Journal of operational Research*, vol. 232, pp. 249-255, 2014.

[34] M. Zeleny, ”A Concept of Compromise Solutions and the Method of the Displaced Ideal”, *Computers and Operations Research*, Vol. 1, No. 3-4, pp. 479-496, 1974.

[35] O. folorunso, R. V. Olufunke and O. Salako, ”An Exploratory Study of Critical Factors Affecting the Efficiency of Sorting Techniques (Shell, Heap and Treap)”, *Anale Seria Informatica*, vol. viii, pp. 163-172, 2010.
[36] O. J. Boxma and J. W. Cohen, ”Heavy Traffic Analysis of the G1/G/1 Queue with Heavy Tailed Distributions”, *Queueing Systems,* vol. 33, pp. 177-204, 1999.
[37] P. E. Drenick and R. F. Drenick, ”A Design Theory for Multi-Processing Computing Systems”, *Large Scale Systems*, vol. 12, pp. 155–172, 1987.
[38] P. Gizegorreswski, ”Nearest Interval Approximation of a fuzzy numbers”, *Fuzzy sets and systems,* vol. 130, pp. 321-330, 2002.
[39] Q. Zhang and H. Li, ”MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition”, *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, vol. 11, no. 6, pp. 712-732, 2007.
[40] R. F. Drenick and K. K. Ko, *Decentralized Control of Street Traffic, Control of Urban Traffic Systems*, Edited by W. S. Levineet al., United Engineering Trustees, New York, New York, 1981.
[41] S. Mouzas, ”Efficiency versus effectiveness in business networks”, *Journal of Business Research*, vol. 59, no. 11, pp. 24-32, 2006.
[42] W. Warid, H. Hizam, N. Mariun and N. Izzri Abdel-Wahab, ”An Efficacious Multi-Objective Fuzzy Linear Programming Approach for Optimal Power Flow Considering Distributed Generation”, *PLoS ONE*, vol. 11, no. 3, 2019.
[43] X. T. Nguyen and V. D. Nguyen, ”Support-Intuitionistic Fuzzy Set: A New Concept for Soft Computing”, *International Journal of Intelligent Systems and Applications,* vol. 4, pp. 11-16, 2015.

Dr. Stephen Gbenga Fashoto is a Senior Lecturer in the Department of Computer Science in University of Swaziland, Kwaluseni, Swaziland. He was the Director of ICT and a Senior Lecturer at Kampala International University, Kampala, Uganda between 2015 and 2016. He was a Post-Doctoral Fellow in the Department of Decision Sciences in the University of South Africa in 2016/2017. He was also a Lecturer in the Department of Computer Science in Redeemer's University, Ede, Osun state, Nigeria between 2007 and 2014. He obtained his B.Sc. (Honours) in Computer Science & Mathematics and M.Sc. in Computer Science both from University of Port Harcourt, Nigeria in 1998 and 2004 respectively and Ph.D. in Computer Science from University of Ilorin, Nigeria in 2014. He has several publications in journals (locally and internationally) and conference proceedings (both locally and internationally). He is a member of Computer Professionals of Nigeria and IAENG. He has over fifteen years of teaching and research experience in the university. His research interests are in Computer Science Education, Machine Learning, Computational Intelligence, Health informatics, Data mining and application of optimization techniques. He is also a winner of several travel grant awards such as POSSE(Google award), Bill and Melinda Gate award, Black in AI (Google award), IndabaX award (Google award) and so on just to mention a few. He is the Chair of Computational Intelligence and Health Informatics Research Group in the Department of Computer Science in University of Swaziland.

Sulaiman Sani is a Senior Lecturer-Mathematics, University of Eswatini (formerly, University of Swaziland) Kwaluseni, Eswatini. He holds a Doctorate Degree in Mathematics of the University of Botswana, Gaborone since 2015. Sani Sulaiman is actively engaged in Mathematics of Finance, Operational Research and Optimization of service systems of Economics, Engineering and Environment to date. He has published several research papers in these areas and was invited by many academic institutions worldwide for presentation of research papers at conferences of International repute including the prestigious African Academy of Sciences Conference of 2016 in Abuja-Nigeria. Sani Sulaiman received the Annual Best Lecturer award for Physical Sciences of the School of Engineering and Applied Sciences, Kampala International University Uganda in December 2018. He is a recipient of the Jalingo local Government, Nigeria merit medal in 2020. He has taught and tutored Mathematics in several African Universities including the University of Botswana, Gaborone. He is a member of several professional associations including the IAENG. Sani Sulaiman has supervised to completion five (5) M.Sc. theses to which three (3) led to the award of M.Sc. Mathematics and two (2) led to the award of M.SEE. Electrical Engineering. He is happily married with four (4) children.