Sperm Motility Algorithm for Solving Fractional Programming Problems under Uncertainty

Osama Abdel Raouf
Operations Research and DSS Department,
FCI, Menoufia University,
Menoufia, Egypt

Bayoumi M. Ali Hassan
Decision Support Department,
Faculty of Computers and Information,
Cairo University, Cairo, Egypt

Ibrahim M. Hezam
Department of Mathematics and Computer
Faculty of Education, Ibb University,
Ibb, Yemen

***Abstract*—This paper investigated solving Fractional Programming Problems under Uncertainty (FPPU) using Sperm Motility Algorithm. Sperm Motility Algorithm (SMA) is a novel metaheuristic algorithm inspired by fertilization process in human, was proposed for solving optimization problems by Osama and Hezam [1]. The uncertainty in the Fractional Programming Problem (FPP) could be found in the objective function coefficients and/or the coefficients of the constraints. The uncertainty in the coefficients can be characterised by two methods. The first method is fuzzy logic-based alpha-cut analysis in which uncertain parameters are treated as fuzzy numbers leading to Fuzzy Fractional Programming Problems (FFPP). The second is Monte Carlo simulation (MCS) in which parameters are treated as random variables bound to a given probability distribution leading to Probabilistic Fractional Programming Problems (PFPP). The two different methods are used to revise the trustiness in the transformation to the deterministic domain. A comparative study of the obtained result using SMA with genetic algorithm and the two SI algorithms on a selected benchmark examples is carried out. A detailed comparison is induced giving a ranked recommendation for algorithms and methods proper for solving FPPU.**

Keywords—Sperm Motility Algorithm; Fractional Programming; Uncertainty; Fuzzy Programming; Monte Carlo Method

I. Introduction

In real life decision-making situations, the decision makers often face problems in making decision from linear/non-linear fractional programming problems (FPPs); the objectives are generally conflicted, non-commensurable and fuzzy in nature and many considerations of the vague nature of uncertainty should be taken in the formulation of the problem. Naturally the objective functions and constraints are uncertainty in their nature and involve many fuzzy or stochastic parameters. In most of the practical situations the possible value of the parameters involved in the objective could not be defined precisely due to the lack of available data. The concept of fuzzy sets seems to be most appropriate to deal with such imprecise data. There are many different algorithms to solve fuzzy fractional programming problem. Many of these approaches are based upon traditional optimization or classical methods. That is, it is still inefficient and lack universality, especially for non-linear and non-differentiable fractional objective functions. However, intelligent optimization techniques, such as evolutionary computation have a growing interest as a problem solver in the field of optimization and computer science. Rezaee, A. [2] proposed an interactive particle swarm optimization for general fuzzy non-linear goal programming. XU, X.L., et al. [3] modified particle swarm optimization algorithm to solve intuitionistic fuzzy integer programming. They convert the fuzzy integer programming into integer programming by membership function and resolved it by improving particle swarm optimization. Yi, L., et al. [4] proposed and analysed fuzzy form of the bi-level programming by using the interactive method and by imposing the improved PSO algorithm. They firstly convert the basic bi-level programming problem into its intuitionistic fuzzy form, which is intuitionistic fuzzy bi-level programming. The membership and non-membership function could drive the integer fuzzy bi-level programming to the global optimum result. An interactive computational method is proposed for obtaining the global optimal solution of integer fuzzy bi-level programming. The method adopts the improved PSO algorithm, by imposing a mechanism to improve the diversity and expand the search space of the particle. Hezam, I.M. et al. [5]–[9] introduced solution for different types of fractional programming problem using metaheuristic algorithms. Abebe, A. et al. [10] presented a comparison between Monte Carlo simulation (MCS) and fuzzy logic-based α-level cut analysis. They tested both techniques on a model of groundwater contamination transport where the decay rate of the contaminant is considered to be uncertain. Cantoni, M. et al. [11] presented an approach to the optimal plant design under conflicting safety and economic constraints, based on the coupling of a Monte Carlo evaluation of plant operation with a genetic algorithms-maximization procedure. Buckley, et al. [12] presented Monte Carlo methods in fuzzy optimization using two methods to handle the uncertainty, (1) Kerre's Method, and (2) Chen's Method. Yeh, W.C. et al. [13] proposed Particle Swarm Optimization (PSO) based on Monte Carlo simulation (MCS), to solve complex network reliability optimization problems. Sar and Kahraman [14] used the fuzzy MCS method to determine the best investment strategy on new product selection for an organization in the condition when the fuzzy net present value is not the only point of concern for decision making. Fan, YR. et al. [15] developed a generalised fuzzy linear programming method for dealing with uncertainties expressed as fuzzy sets. The feasibility of fuzzy solutions of the generalised fuzzy linear programming problem was investigated. A stepwise interactive algorithm based on the idea of the design of the experiment is then introduced to solve the generalised fuzzy linear programming problem. A comparison between the solutions obtained through the stepwise interactive algorithm and Monte Carlo method is finally conducted to demonstrate the robustness of the stepwise interactive algorithm method.

The purpose of the current work is to solve fractional programming problems using Sperm Motility Algorithm under uncertainty. While the uncertainty is characterised using two different methods; the α-level set fuzzy number based method and the Monte Carlo method. Throughout the literature review, FPP under uncertainty have never been solved by metaheuristic algorithms. The Monte Carlo method is used also for the first time in handling the uncertainty in the coefficients of the FPP.

The remainder of this paper is organised as: Section 2 introduce the problem statement and solution concepts. Monte Carlo method is reviewed in Section 3. In Section 4, an overview of Sperm Motility Algorithm (SMA) is introduced. In Section 5, the proposed algorithms for FPPU is discussed. In Section 6, numerical examples with discussion are introduced. Finally, Section 7 is the concluding part of the paper.

II. Problem Statements and Solution Concepts

In this paper, the general mathematical model of the FPPU is as follows:

$$\text{min/ max} \quad z(x_1,...,x_n)=\sum_{i=1}^{p}\frac{\tilde{f}_i(x)}{\tilde{g}_i(x)} \tag{1}$$

$$\textit{subject to} \quad x\in S, x\geq 0$$

$$S=\left\{x\in R^n \left| \begin{array}{l}\tilde{h}_k(x)\geq 0, \quad k=1,...,K;\\ \tilde{m}_j(x)=0, \quad j=1,...,J;\\ \tilde{x}_i^l\leq x_i\leq\tilde{x}_i^u, \; i=1,..,n.\end{array}\right.\right\}$$

$$\tilde{g}_i(x)\neq 0, \; i=1,..,n.$$

where, $\tilde{f}_i(x)$, $\tilde{g}_i(x), \tilde{h}_k(x), and\ \tilde{m}_j(x)$, are supposed to be continuous functions, with fuzzy coefficients. S is compact.

~ represents the presence of fuzzy numbers within the matrices or vectors. It's obvious that the uncertainty appears in the coefficients of the objective function and/or the coefficients of constraints.

A. Definition 1

[16] A real fuzzy number $\tilde{J}$ is a continuous fuzzy subset from the real line R whose triangular membership function $\mu_{\tilde{J}}(J)$ is defined by a continuous mapping from R to the closed interval [0,1], as shown in Figure 1, where,

(1) $\mu_{\tilde{J}}(J)=0$ for all $J\in(-\infty,a_1]$,
(2) $\mu_{\tilde{J}}(J)$ is strictly increasing on $J\in[a_1,m]$,
(3) $\mu_{\tilde{J}}(J)=1$ for $J=m$,
(4) $\mu_{\tilde{J}}(J)$ is strictly decreasing on $J\in[m,a_2]$,
(5) $\mu_{\tilde{J}}(J)=0$ for all $J\in[a_2,+\infty)$.

This will be elicited by:

$$\mu_{\tilde{J}}(J)=\begin{cases}0, & J\leq a_1,\\ \dfrac{J-a_1}{m-a_1}, & a_1\leq J\leq m,\\ \dfrac{a_2-J}{a_2-m}, & m\leq J\leq a_2,\\ 0, & J\geq a_2.\end{cases} \tag{2}$$

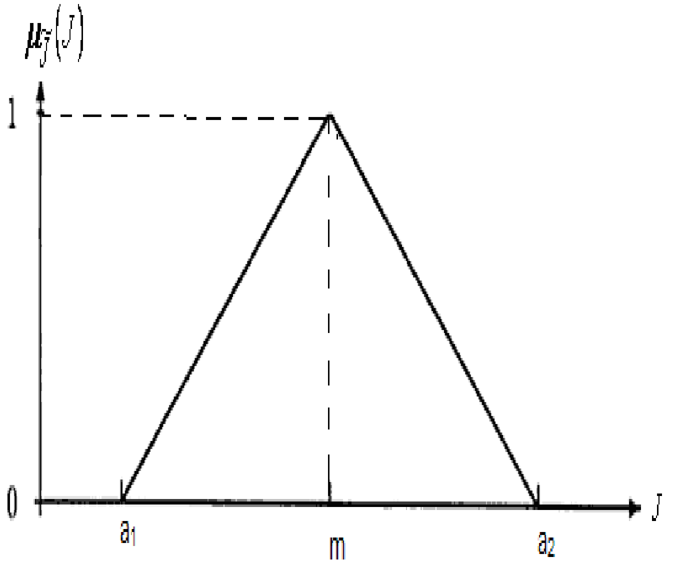

Fig. 1. Membership Function of Fuzzy Number J.

where, m is a given value, a_1 and a_2 denote the lower and upper bounds. Sometimes, it is more convenient to use the notation explicitly highlighting the membership function parameters. In this case, we obtain

$$\mu(J;a_1,m,a_2)=\text{Max}\left\{\text{Min}\left[\frac{J-a_1}{m-a_1},\frac{a_2-J}{a_2-m}\right],0\right\}$$

In what follows, the definition of the α-level set or α-cut of the fuzzy number $\tilde{J}$ is introduced.

B. Definition 2

[16] The α-level set of the fuzzy parameters $\tilde{J}$ in problem (1) is defined as the ordinary set $L_\alpha(\tilde{J})$ for which the degree of membership function exceeds the level, α, $\alpha\in[0,1]$, where:

$$L_\alpha(\tilde{J})=\{J\in R|\mu_{\tilde{J}}(J)\geq\alpha\}$$

For certain values α_j^* to be in the unit interval, the problem (FPPU) (1) can be reformulated as in the following non-fuzzy optimization model (α-FPP):

$$\text{min/ max} \quad z(x_1,...,x_n)=\sum_{i=1}^{p}\frac{f_i(x)}{g_i(x)} \tag{3}$$

$$subject\ to\ \ x \in S, x \geq 0$$

$$S = \left\{ x \in R^n \middle| \begin{array}{l} h_k(x) \geq 0, \quad k = 1,...,K; \\ m_j(x) = 0, \quad j = 1,...,J; \\ x_i^l \leq x_i \leq x_i^u, \ i = 1,..,n. \end{array} \right\}$$

$$g_i(x) \neq 0, \ i = 1,..,n.$$

$\mu_{\tilde{j}} \geq \alpha_j$ *,where j is any* coefficient

Problem (α-FPP) (2) can be rewritten as:

$$\min/\max \quad z(x_1,...,x_n) = \sum_{i=1}^{p} \frac{f_i(x)}{g_i(x)} \tag{4}$$

$$subject\ to\ \ x \in S, x \geq 0$$

$$S = \left\{ x \in R^n \middle| \begin{array}{l} h_k(x) \geq 0, \quad k = 1,...,K; \\ m_j(x) = 0, \quad j = 1,...,J; \\ x_i^l \leq x_i \leq x_i^u, \ i = 1,..,n. \end{array} \right\}$$

$$g_i(x) \neq 0, \ i = 1,..,n.$$

$$j_l \leq j^{\alpha} \leq j_u$$

where j_l, j_u are lower and upper bounds on *j*, where the *j* α means the value of *j* at α^0, $\alpha \in [0,1]$.

III. Monte Carlo Method

The MCS technique is an especially useful means of analyzing situations involving risk to obtain approximate answers when a physical experiment or the use of analytical approaches is either too burdensome or not feasible [14]. Monte Carlo methods vary but tend to follow a particular pattern. Using the MC method starts with defining a domain of possible inputs, then the inputs are generated randomly from a probability distribution over the domain. In this work, we used uniform distribution in order to perform fast deterministic computation on the inputs and aggregate the results. The shape of the membership function used in the α-cut fuzzy method is the same as the shape of the probability density function used in the MCSs.

IV. Overview of SMA

Sperm Motility Algorithm [1] is an evolutionary algorithm inspired by the fertilization process in human. During the search process, there are mainly several principle rules. (1) All sperms are attracted toward ovum of their species chemoattractant. (2) Attractiveness is proportional to chemoattractant concentration and these both increase whenever the sperm is close to the ovum. (3) The best healthy or highest quality of sperm -type A- will be carried over to the next generations; other less quality sperms -types B, C and D are neglected with a probability $P_a \in [0\ 1]$. (4) One sperm penetrates the ovum, and this rule can be modified to suit the multi-objective optimization as there can be more than one egg (such as fraternal twins). (5) More than 250 million sperms swim randomly with the velocity v_i at position x_i forward to the ovum, where motility can be described by the Stokes equations.

The mathematical modelling of sperm motility is considered by Stokes equation:

$$Re\left(\frac{\partial v}{\partial t} + v \cdot \nabla v\right) + \nabla p = \mu \nabla^2 v + f \tag{5}$$

$$\nabla \cdot v = 0 \ \ x \in \Omega$$

where, *p* is the pressure, including the gravitational potential. μ is kinematic viscosity and *f* is the force density. *v* is the velocity vector field in the domain *Ω*. For a micro swimmer such as a sperm, *Re* is approximately 0.01. That means Stokes equation a linearised form of the Navier–Stokes equations in the limit of small Reynolds number, and the inertial terms in the NS equation can be omitted to obtain the simpler Stokes equation:

$$\nabla p = \mu \nabla^2 v + f \tag{6}$$

$$\nabla \cdot v = 0 \ \ x \in \Omega$$

The velocity solution corresponding to this fundamental singularity is given by:

$$v_i(t) = \left(\frac{1}{8\pi\mu}\right)\left(\frac{\delta_{ij}}{r} + \frac{r_i r_j}{r^3}\right)F_j$$

$$= \left(\frac{1}{8\pi\mu}\right) S_{ij}(x,\xi) F_j\,;\, i,j = 1,2,3. \tag{7}$$

where the $S_{ij}(x,\xi)$ is known as the Stokeslet, or Oseen-Burgers tensor, δ is Dirac delta distribution centered at ζ. The flow is due to a force F_j concentrated at the point ζ, and

$$r_i = x - \xi$$

$$r^2 = r_1^2 + r_2^2 + r_3^2.$$

The position is updated as follow:

$$x_{i+1}(t) = x_i(t) + \left(\frac{\delta t}{2}\right)\left(v_{i+1}(t) + v_i(t)\right) + \beta\left(x_i(t) - g^*\right) \tag{8}$$

Non-linear spatial chemoattractant concentration gradient field is as follow:

$$c_i(t) = c_0(t) + c_1\left(\|g^* - x_i(t)\|\right)^{-b} \tag{9}$$

where, *c(t)* is the concentration, *x(t)* is the position, c_1 and *b* are the proportion coefficient and the power of the major term position, respectively. c_0 represent the remaining terms. g^* is the current best solution found among all solutions at the current generation/iteration.

The basic steps of the SMA can be summarised as the pseudo code shown below:

```
Algorithm 1: The original Sperm Motility Algorithm
Begin
Define objective function $f(x), x = (x_1, x_2, \dots, x_d)^T$
initialise N sperm population size
generate initial position $x_0$ and velocity $v_0$ and initial concentration $c_0$ of N sperm of N sperm
define all SMA parameters ($c_0$, $\beta$, $\mu$…etc.).
while (t< Maximum Generation) or (stopping criterion);
    for i=1: N do
        calculate velocity $v_i$ from data at $t = t_i$; equation (7);
        update position $x_i$ for sperm i from equation (8);
        evaluate each sperm individual according to its position.
        if new solution is better, update it in the population;
        calculate $c_i$ from equation (9).
        if $c_i \le c_{i-1}$ then neglect [Abandon a fraction ($P_a$) of worse sperm];
        Check constraints satisfactions.
    end for
Sort the population/sperm from best to worst and find the current best.
end while
Post-processing the results and visualization.
End
```

The constraints are handled using the same rules in [1].

V. Proposed Procedures for FPPU

In this section, we suggest two procedures to solve FPPU. The suggested procedures can be summarised as follows:

Procedure I: *Sperm Motility Algorithm for FFPP based on the α Level Set:*

Step1: Start with initial level set $\alpha^\circ = \left(\alpha_j^\circ\right)$ randomly chosen from the interval [0, 1].

Step 2: Determine the points $\left(a_1, m, a_2\right)$ corresponding to the coefficient numbers in the objective function and the constraints to elicit membership functions $\mu_{\tilde{j}}\left(j\right)$.

Step 3: Determine the lower and upper bounds for all coefficient numbers at each α- level cut.

Step 4: Choose certain values for all $j^* \in \left[j_l, j_u\right]$ corresponding to the α - level cut $\alpha = \alpha^* \in \left[0,1\right]$.

Step 5: Convert the given problem (1) into its non-fuzzy form (α-FPP) problem (4).

Step6: Use SMA to solve problem (4). The obtained solution is a near optimal solution for the original FPPU model.

Step 7: Set $\alpha = \left(\alpha^\circ + \text{step}\right),\ \alpha \in \left[0,1\right]$.

Step 8: Go to step (1) with a new α until the interval [0,1] is fully exhausted. Then, stop.

Procedure II: *Sperm Motility Algorithm for PFPP based on Monte Carlo Method:*

Step 1: Define the objective function and the constraints.

Step 2: Determine the vector interval for all uncertainty coefficient in the objective functions and/or the constraints.

Step 3: Employ Monte Carlo method to generate random numbers from the uniform distribution.

Step 4: Use SMA to solve the deterministic problem.

Step 5: Termination checking. Repeat Steps 3 and 4 until definite termination conditions are met.

VI. Illustrative Examples with Discussion

Benchmark examples were collected from literature to demonstrate the efficiency and robustness of the proposed algorithms in solving FPPU. The numerical results of the four used algorithms are compared among the two methods used for handling the uncertainty illustrated in Tables 1 to 6. The algorithms have been implemented by MATLAB R2011 on core (TM) i3 to 2.27 GHz processor.

A. Example 1

$$f_1:\ \min\ z = \frac{\langle 1,2,3\rangle \sin x^2 + \langle 1,3,5\rangle e^y}{\langle 1,3,5\rangle e^y + \langle 1,2,3\rangle \sin y}$$

$$\textit{subject to}\quad \langle 1,2,3\rangle \le x, y \le \langle 3,4,5\rangle;$$

Set $\alpha = \alpha^* \in [0,1]$ with the following membership functions to convert the above fuzzy problem (FFPP) into its non-fuzzy version refer to problem (2).

Let also the fuzzy parameters $\tilde{J}$ given by the following fuzzy numbers listed in the table below:

α-level set	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$
$\langle 1,2,3\rangle$	[1, 3]	[1.5, 2.5]	2
$\langle 1,3,5\rangle$	[1, 5]	[2, 4]	3
$\langle 3,4,5\rangle$	[3, 4]	[3.5, 4.5]	4

Choose the certain values for all $j^* \in \left[j_l, j_u\right]$ corresponding to the α- level cut, α=α * = [0, 1]. Now, the fuzzy problem (FFPP) is converted to the non-fuzzy version (α −FPP) as in the following form:

$$\alpha = 0$$

$$f_1^{\alpha_0}:\ \min\ z = \frac{\sin x^2 + e^y}{e^y + \sin y}$$

$$\textit{subject to}\quad 1 \le x, y \le 3;$$

$$\alpha = 0.5$$

$$f_1^{\alpha_{0.5}}:\ \min\ z = \frac{1.5\sin x^2 + 2e^y}{2e^y + 1.5\sin y}$$

$$\textit{subject to}\quad 1.5 \le x, y \le 3.5;$$

$$\alpha = 1$$

$$f_1^{\alpha_1}:\ \min\ z = \frac{2\sin x^2 + 3e^y}{3e^y + 2\sin y}$$

$$\textit{subject to}\quad 2 \le x, y \le 4;$$

After applying the SMA algorithm to solve problems $f_1^{\alpha_0}$, $f_1^{\alpha_{0.5}}$, and $f_1^{\alpha_1}$, the obtained solution is the near optimal solution of the original FFPP.

TABLE I. Comparison Results of the SMA, PSO, FA and GA on F_1 Based on Fuzzy A - Level Cut

Fun. / Tec.	Num. of Iteration	PSO		FA		SMA		GA	
		Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
$\alpha = 0$	30	(2.261,1) z=0.17735	1.96	(3,1.16) z=0.1758	0.413	(3,1) z=0.15177	0.111	(2.257,1) z=0.17734	0.28
	40	(2.292,1) z=0.17831	2.8	(3,1) z=0.15177	0.331	(3,1) z=0.15177	0.137	(2.999,1.002) z=0.152	0.29
	50	(2.285,1) z=0.17798	3.56	(3,1) z=0.15177	0.437	(3,1) z=0.15177	0.168	(2.999,1.0003) z=0.1518	0.294
	60	(2.266,1) z=0.17745	4.31	(3,1) z=0.15177	0.558	(3,1) z=0.15177	0.204	(2.999,1.0002) z=0.151798	0.357
$\alpha = 0.5$	30	(3.462,1.5) z=0.1274	1.93	(3.446,1.5) z=0.127385	0.28	(3.446,1.5) z=0.127385	0.068	(3.446,1.5) z=0.127386	0.297
	40	(3.45,1.5) z=0.1274	2.65	(3.446,1.5) z=0.127385	0.397	(3.446,1.5) z=0.127385	0.077	(3.451,1.5) z=0.127394	0.308
	50	(3.462,1.5) z=0.1274	3.18	(3.446,1.5) z=0.127385	0.45	(3.446,1.5) z=0.127385	0.124	(3.446,1.5) z=0.127386	0.35
	60	(3.45,1.5) z=0.12739	3.91	(3.446,1.5) z=0.127385	0.544	(3.446,1.5) z=0.127385	0.18	(3.4496,1.5) z=0.127388	0.353
$\alpha = 1$	30	(4,2) z=0.13249	2.34	(4,2) z=0.13249	0.26	(4,2) z=0.13249	0.1	(3.999,2.0001) z=0.13268	0.293
	40	(4,2) z=0.13249	3.32	(4,2) z=0.13249	0.38	(4,2) z=0.13249	0.12	(3.999,2.0004) z=0.132501	0.301
	50	(4,2) z=0.13249	4.11	(4,2) z=0.13249	0.47	(4,2) z=0.13249	0.13	(3.999,2.0005) z=0.132566	0.313
	60	(4,2) z=0.13249	4.68	(4,2) z=0.13249	0.61	(4,2) z=0.13249	0.14	(3.999,2.0002) z=0.13276	0.335

TABLE II. Solution Results using SMA, PSO, FA, and GA on F_1 Characterised by Monte Carlo Method

	PSO		FA		SMA		GA	
	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
f^{mean}	(3.9597, 1.983) z=0.130724125	6.3	(3.9597, 1.983) z=0.1307241	0.897	(3.9597, 1.983) z=0.130724125	0.215	(3.9597, 1.983) z=0.130724125	0.31
f^{min}	(2.259, 1.003) z=0.17617	6.8	(3.0202, 1.003) z=0.14204	1.07	(3.0202, 1.003) z=0.14204	0.27	(3.020, 1.0038) z= 0.142144	0.3234
f^{max}	(4.95, 2.999) z=0.1357536	7.1	(3.999, 2.999) z=0.145	0.873	(4.942, 2.999) z=0.13575	0.295	(4.9414, 2.999) z=0.135779	0.3046

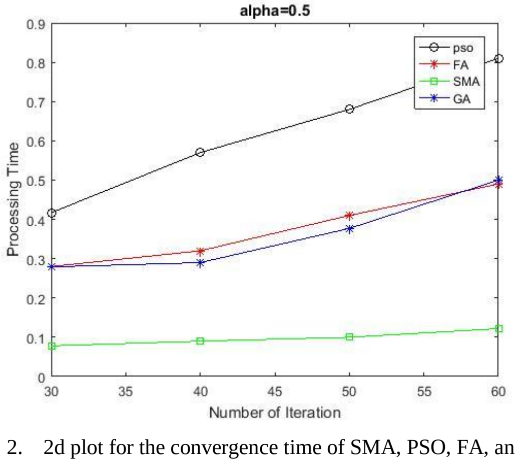

Fig. 2. 2d plot for the convergence time of SMA, PSO, FA, and GA

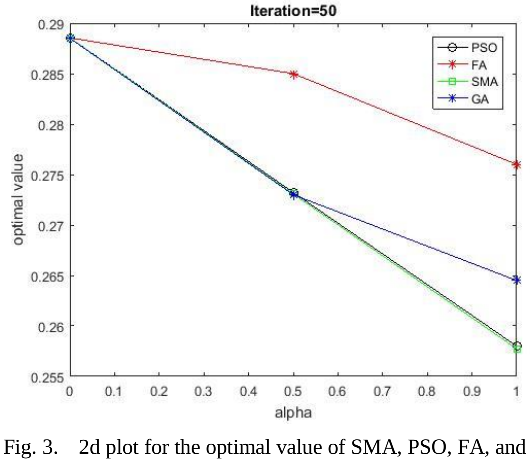

Fig. 3. 2d plot for the optimal value of SMA, PSO, FA, and GA

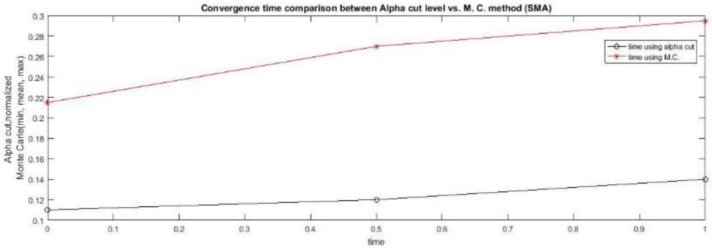

Fig. 4. Convergence time comparison between α- cut level vs. Monte Carlo method

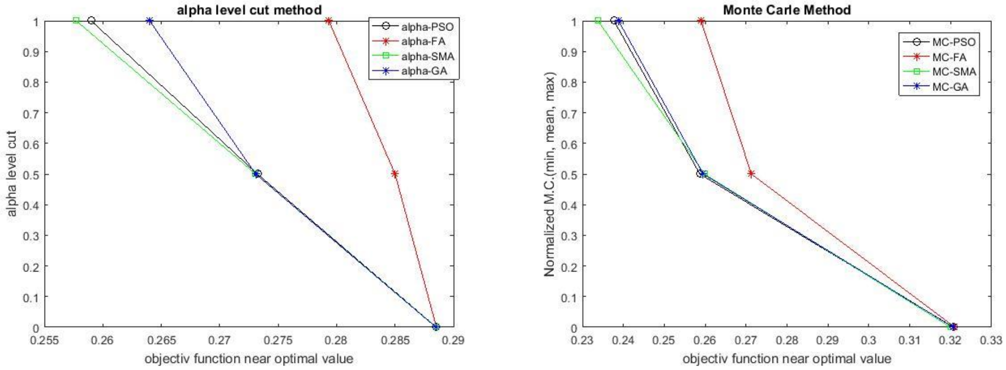

Fig. 5. Comparison results obtained objective function f_1 value based on α level cut and Monte Carlo method

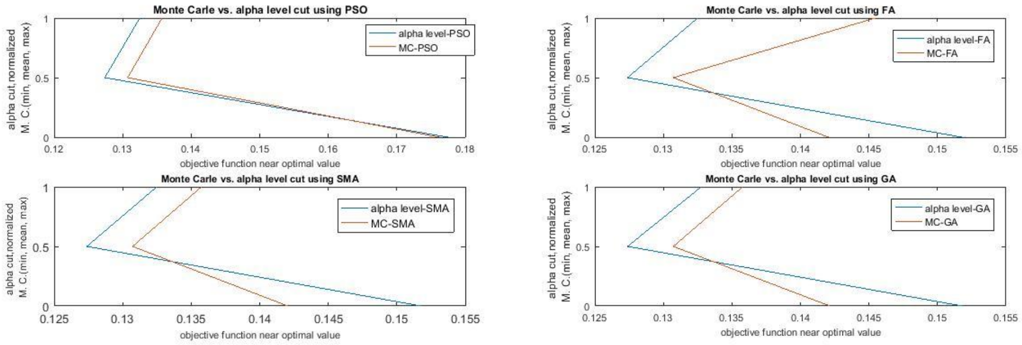

Fig. 6. α- cut level vs. MC solution result for the SMA, PSO, FA, and GA

B. Example 2

$$f_2:\ \min\ z=\frac{\langle 2,5,8\rangle x+\langle 1,3,5\rangle y}{\langle 2,5,8\rangle x+\langle 1,2,4\rangle y+\langle 1,1.5,2\rangle}$$

$$\textit{subject to}\quad \langle 0,0.5,1\rangle \le x \le \langle 1,2,3\rangle;$$

$$\langle 0,0.5,1\rangle \le x \le \langle 1,3,5\rangle;$$

TABLE III. COMPARISON RESULTS OF THE SMA, PSO, FA, AND GA ON F_2 BASED ON FUZZY α - LEVEL CUT

Fun. / Tec.	Num. of Iteration	PSO		FA		SMA		GA	
		Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
$\alpha = 0$	30	(0,0) z=0	2.036	(0,0) z=0	0.321	(0,0) z=0	0.052	(0,0) z=0	0.33
	40	(0,0) z=0	2.697	(0,0) z=0	0.44	(0,0) z=0	0.091	(0.0002,0) z=0.00005	0.34
	50	(0,0) z=0	3.321	(0,0) z=0	0.492	(0,0) z=0	0.116	(0,0) z=0	0.39
	60	(0,0) z=0	3.91	(0,0) z=0	0.556	(0,0) z=0	0.126	(0,0) z=0	0.49
$\alpha = 0.5$	30	(0.25,0.25) z=0.55	2.154	(0.25,0.25) z=0.55	0.28	(0.25,0.25) z=0.55	0.087	(0.25,0.25) z=0.55	0.283
	40	(0.25,0.25) z=0.55	2.885	(0.25,0.25) z=0.55	0.33	(0.25,0.25) z=0.55	0.089	(0.25003,0.25) z=0.43	0.343
	50	(0.25,0.25) z=0.55	3.288	(0.25,0.25) z=0.55	0.47	(0.25,0.25) z=0.55	0.095	(0.25,0.25) z=0.55	0.349
	60	(0.25,0.25) z=0.55	4.259	(0.25,0.25) z=0.55	0.62	(0.25,0.25) z=0.55	0.113	(0.25003,0.2502) z=0.43	0.386
$\alpha = 1$	30	(0.5,0.5) z=0.8	1.96	(0.5,0.5) z=0.8	0.264	(0.5,0.5) z=0.8	0.06	(0.5,0.5) z=0.8	0.285
	40	(0.5,0.5) z=0.8	2.53	(0.5,0.5) z=0.8	0.343	(0.5,0.5) z=0.8	0.11	(0.5005,0.5) z=0.8001	0.29
	50	(0.5,0.5) z=0.8	3.32	(0.5,0.5) z=0.8	0.498	(0.5,0.5) z=0.8	0.15	(0.5,0.5) z=0.8	0.32
	60	(0.5,0.5) z=0.8	3.9	(0.5,0.5) z=0.8	0.549	(0.5,0.5) z=0.8	0.19	(0.5,0.5) z=0.8	0.35

TABLE IV. SOLUTION RESULTS USING SMA, PSO, FA, AND GA ON F_2 CHARACTERISED BY MONTE CARLO METHOD

	PSO		FA		SMA		GA	
	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
f_{mean}	(0. 496, 0. 472) z=0.70678	6.533	(0. 496, 0. 472) z=0.70678	1.128	(0. 496, 0. 472) z=0.70678	0.25	(0. 496, 0. 472) z=0.70678	0.2979
f_{min}	(0.008, 0.0017) z=0.0191036	7.55	(0.008, 0.0017) z=0.0191036	1.02	(0.008, 0.0017) z=0.0191036	0.252	(0.008, 0.0017) z=0.0191036	0.3233
f_{max}	(0.985, 0.998) z=0.92125	6.77	(0.985, 0.998) z=0.92125	0.98	(0.985, 0.998) z=0.92125	0.27	(0.985, 0.998) z=0.92125	0.294

C. Example 3

$$f_3:\ \min\ z = \frac{\langle 0,1,2\rangle \sin\left(x^2 - y^2\right) + \langle 1,3,5\rangle}{\langle 1,2,3\rangle \log\left(x^2 + y^2\right)}$$

$$subject\ to\quad \langle 1,2,3\rangle \le x\,,y \le \langle 4,5,6\rangle;$$

TABLE V.

TABLE VI. SOLUTION RESULTS USING SMA, PSO, FA, AND GA ON F_3 CHARACTERISED BY MONTE CARLO METHOD

	PSO		FA		SMA		GA	
	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
f_{mean}	(4.83, 4.9799) z=0. 2587	15.1	(3.999, 4.885) z=0.2713	0.985	(4.84, 4.9799) z=0.25984	0.325	(4.777, 494) z=0.2595	0.29
f_{min}	(3.988, 4.045) z=0.3207	5.7	(3.99, 4.045) z=0.321	1.14	(3.997, 4.045) z=0.321	0.314	(3.995, 4.044) z= 0.3206	0.287
f_{max}	(5.9977, 5. 6) z=0.2377	25.2	(3.999, 5. 49) z=0.259	0.95	(5.84, 5.99) z=0.2338	0.253	(5.59, 5.735) z=0.238811	0.295

TABLE VII. COMPARISON RESULTS OF THE SMA, PSO, FA, AND GA ON F_3 BASED ON FUZZY A - LEVEL CUT

Fun. / Tec.	Num. of Iteration	PSO		FA		SMA		GA	
		Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)	Optimal value	Time (Sec.)
$\alpha = 0$	30	(4,4) z=0.288539	1.65	(4,4) z=0.288539	0.224	(4,4) z=0.288539	0.076	(3.999,3.996) z=0.2886	0.247
	40	(4,4) z=0.288539	2.41	(4,4) z=0.288539	0.311	(4,4) z=0.288539	0.089	(3.999,3.999) z=0.2885	0.3
	50	(4,4) z=0.288539	2.73	(4,4) z=0.288539	0.448	(4,4) z=0.288539	0.11	(3.999,3.999) z=0.28854	0.31
	60	(4,4) z=0.288539	3.99	(4,4) z=0.288539	0.489	(4,4) z=0.288539	0.136	(3.999,3.996) z=0.2885	0.44
$\alpha = 0.5$	30	(4.3133,4.46) z=0.2757	4.17	(3.32,4.34) z=0.294	0.284	(4.26,4.47) z=0.2781	0.078	(4.29,4.47) z=0.274	0.2795
	40	(4.5,3.05) z=0.2956	5.71	(3.8,4.2) z=0.2921	0.32	(4.32,4.5) z=0.273127	0.09	(4.32,4.49) z=0.27317	0.29
	50	(4.32,4.5) z=0.27323	6.81	(3.98,4.2) z=0.285	0.41	(4.32,4.5) z=0.273076	0.1	(3.5,4.497) z=0.287	0.3776
	60	(3.51,4.48) z=0.288	8.11	(3.5,4.5) z=0. 2245	0.496	(4.325,4.5) z=0.27307	0.122	(4.3248,4.499) z=0.273075	0.51
$\alpha = 1$	30	(4.2,5) z=0.2685	6.09	(3.45,4.44) z=0.2896	0.234	(4.84,5) z=0.257729	0.07	(4.84,.999) z=0.257718	0.282
	40	(4.5,4.7) z=0.26662	8.36	(3.8,4.7) z=0.2762	0.316	(4.84,5) z=0.257729	0.08	(4.8,4.966) z=0.25866	0.318
	50	(4.8,5) z=0.258	10.7	(3.66,4.61) z=0.2821	0.415	(4.84,5) z=0.257729	0.122	(4.93,4.43) z=0.26445	0.33
	60	(4.8,4.9) z=0.259	12.9	(3.3,5) z=0.27933	0.538	(4.84,5) z=0.257729	0.13	(4.954,4.43) z=0.264	0.35

From the solution results of the three selected benchmark examples, some observation could be noticed. The comparison is carried among these possible solution strategies using four algorithms along with two uncertainty characterizing methods. Figure 2 shows the advantages of the SMA algorithm among the rest three algorithm, where PSO come last convergence time. The comparison was held using the same uncertainty characterizing methods. Figure 3 shows the advantages of the SMA algorithm among the rest three algorithm, it found that as the alpha-cut value increases, the optimal value is improving. The comparison was held using the same uncertainty characterizing methods. Figure 4 is a comparison based on the same solution algorithm but this time using two uncertainty characterizing methods which shows a superiority for the α-level cut fuzzy logic over the Monte Carlo method with respect to computational time. Figure 5 shows the solution results using fuzzy logic where all the four used algorithms gave almost the same near optimal solution expected at $\alpha = 0$. Figure 6 show a slight difference in the objective function value using α-cut vs. Monte Carlo method.

VII. CONCLUSIONS

Sperm motility algorithm was used to solve Fractional Programming Problems under uncertainty (FPPU) and comparing with three algorithms (GA, FA, and PSO) managed to converge to a near optimal solution. Two different methods were used to characterise the uncertainty in the coefficients of the objective function and/or the constraints. The two used methods (fuzzy α level cut and Monte Carlo method) were used alternatively along with the four metaheuristic algorithms generating eight different solution strategies.

A set of comparison was carried out among these different solution strategies respecting the solution of three benchmark examples. The comparative study among the solutions gave a clear indication for the superiority of SMA in converge time. Then comes GA, FA and PSO, respectively as indicated from the results. The SMA algorithm is firstly ranked again in terms of the obtained near optimal solution. However, a slight difference in the optimal solution could be noticed especially in non-linear functions. The α- level cut fuzzy number based method obtained a better optimised solution result with a notable saving in computational time.

REFERENCE

[1] O. A. R. Ibrahim Hezam, “Sperm Motility Algorithm: A Novel Metaheuristic Approach for Global Optimization,” Int. J. Oper. Res., vol. 28, no. 2, pp. 143–163, 2017.

[2] A. REZAEE, “PSO for Fuzzy Goal Programming,” Appl. Comput. Math, 2006.

[3] X. XU, Y. LEI, and W. DAI, “Intuitionistic fuzzy integer programming based on improved particle swarm optimization,” J. Comput. Appl., 2008.

[4] L. Yi, L. Wei-min, and X. Xiao-lai, “Intuitionistic Fuzzy Bilevel Programming by Particle Swarm Optimization,” Comput. Intell., 2008.

[5] O. Raouf and I. Hezam, “Solving Fractional Programming Problems based on Swarm Intelligence,” J. Ind. Eng. Int., 2014.

[6] M. Abdel-Baset and I. Hezam, “An improved flower pollination algorithm for ratios optimization problems,” Applied, 2015.

[7] I. Hezam and O. Raouf, “Employing three swarm intelligent algorithms for solving integer fractional programming problems,” Int. J. Sci., 2013.

[8] I. Hezam and O. Raouf, “Particle swarm optimization approach for solving complex variable fractional programming problems,” Intern J Eng, 2013.

[9] I. Hizam, O. Raouf, and M. Hadhoud, “Solving Fractional Programming Problems Using Metaheuristic Algorithms Under Uncertainty,” Int. J. Adv. Comput., 2013.

[10] A. Abebe and V. Guinot, “Fuzzy alpha-cut vs. Monte Carlo techniques in assessing uncertainty in model parameters,” Proc. 4th, 2000.

[11] M. Cantoni, M. Marseguerra, and E. Zio, “Genetic algorithms and Monte Carlo simulation for optimal plant design,” Reliab. Eng. Syst. Saf., 2000.

[12] and L. J. J. Buckley, James J., Monte Carlo methods in fuzzy optimization. Berlin: Springer, 2008.

[13] W. Yeh, Y. Lin, and Y. Chung, “A particle swarm optimization approach based on Monte Carlo simulation for solving the complex network reliability problem,” IEEE Trans., 2010.

[14] İ. Uçal Sarı and C. Kahraman, “New Product Selection Using Fuzzy Linear Programming and Fuzzy Monte Carlo Simulation,” 2012, pp. 441–448.

[15] Y. Fan, G. Huang, and A. Yang, “Generalized fuzzy linear programming for decision making under uncertainty: Feasibility of fuzzy solutions and solving approach,” Inf. Sci. (Ny)., 2013.

[16] O. Saad, B. Hassan, and I. M. Hezam, “Optimizing the underground water confined steady flow using a fuzzy approach,” African J., 2011.