APPENDIX –B
A modified NSGA-II for FRMOOP: An overview

B.1 Introduction

NSGA (Non-Dominated Sorting in Genetic Algorithms) was developed and proposed by Prof. Deb with his students in 1994 [168] for multiobjective optimization. The algorithm aims to determine the tradeoff surface, which is a set of nondominated solution points, known as Pareto-optimal (PO) or noninferior solutions. The basic idea behind NSGA is the ranking process executed before the selection operation. This process identifies nondominated solutions in the population, at each generation, to form nondominated fronts, based on the concept of nondominance criterion as explained by definitions 8.2.1-8.2.4 in chapter 8. After this, the selection, crossover, and mutation usual operators are performed.

From the past decade the NSGA [18,19] has established as a popular and efficient algorithm for solving MOOP. The algorithm has set certain landmarks in the area and successfully been applied to a variety of engineering design problems that involve multiple objectives. Despite of its effectiveness it is criticized due to the high computational complexity of nondominated sorting, lack of elitism. Also the algorithm generally faces difficulty in achieving diversity in solutions. To overcome the demerits of NSGA an improved version of NSGA known as NSGA-II [19] was introduced that alleviates all the demerits of NSGA with some additional features and results in more diversified solution set with lesser computational complexity.

B.2 Modified NSGA-II

Fuzzy relational multiobjective optimization problem (FRMOOP) presents a class of optimization problems with classical objectives and decision space designed by a system
of fuzzy relational constraints. Due to the special structure of the decision space, general metaheuristics used to solve multiobjective optimization problems cannot be applied in their original form to solve FRMOOP as they might result in infeasible solutions. So the case is with NSGA-II. For this we adopt the algorithm with some modifications that are required to be made in the algorithm not in the operations. The modified version of NSGA-II that we have used presents the combination of elements of original NSGA-II with the repair algorithm that has been introduced to design a feasible recombination operator so as to keep the newly generated solutions feasible.

The solution set of the problem has been introduced in section 3.4.1 of chapter 3. It is clear from the discussion of FRE in chapter 3 that in case of FRE the variables cannot assume values in arbitrary range. To begin the algorithm the initialization described in section 8.4.1 of chapter 8 is used that avoids the unnecessary exploration of the search space. Once the population of finite size has been generated the algorithm preserves original elements of NSGA-II except the recombination operators for the sake of feasibility of solutions; as the general real coded genetic algorithm operators in NSGA-II do not produce the feasible individuals at the end. The original elements of the NSGA-II are kept intact except the recombination operators. The genetic operators, simulated binary crossover and polynomial mutation as used in NSGA-II are applied to the individuals then the infeasible individuals are repaired with the Algorithm 1 described in section 8.4 of chapter 8. The introduction of the new operators does not much affect the efficiency of the algorithm; as the structure of the algorithm remains same. Basically, the introduced repair algorithm is the key element that adopts the algorithm to our domain or application. The whole algorithm of modified NSGA-II is shown in figure B.1.
Figure B.1: Flow chart of Modified NSGA-II
B.3 Illustrative example

In this section, results of the proposed genetic algorithm for multiple linear and nonlinear optimization problems are discussed. We consider an example of multiobjective linear and nonlinear optimization problems and system of fuzzy relational equations with product t-norm based compositions to investigate the nature of the solutions obtained using the proposed procedure.

Example B.1. Consider a four dimensional problem with randomly generated fuzzy matrices $A$ and $b$ as follows:

$$
A = \begin{bmatrix}
0.0986 & 0.8471 & 0.5250 & 0.2460 \\
0.1197 & 0.5436 & 0.8444 & 0.9309 \\
0.5695 & 0.7093 & 0.0083 & 0.9550 \\
0.5333 & 0.5834 & 0.6204 & 0.8944
\end{bmatrix}
$$

$$
b = [0.5139 \ 0.8265 \ 0.7817 \ 0.8618]
$$

The maximum solution is obtained as $[0.9757 \ 0.9258 \ 0.9024 \ 0.9636]$. The values of $x_1$ and $x_2$ of all solution vectors have to be fixed at 0.9757 and 0.9258, respectively. Therefore, we can focus on $x_3$ and $x_4$ only. Since the problem is reduced as a two-dimensional problem, the result can be presented graphically. The test results for some multiple linear and nonlinear optimization problems with this system of fuzzy relation equations as constraints are discussed below.

Case1: Min $f_1(x) = -0.6x_1 + 0.5x_2 + 0.1x_3 + 0.3x_4,$

$\quad f_2(x) = 0.8x_1 - 0.4x_2 + 0.2x_3 - 0.3x_4.$

Figure B.2 shows the Pareto front and the Pareto optimal solutions obtained with the modified NSGA-II.
Figure B.2: PF and Pareto optimal solutions obtained -Example B.1-Case-1

Case 2: \( \begin{align*}
    f_1(x) &= 10(x_3 - 0.5)^2 + 10(x_4 - 0.5)^2 + 5, \\
    f_2(x) &= 10(x_3 - 0.7)^2 + 10(x_4 - 0.7)^2 + 5.
\end{align*} \)

Figure B.3 shows the Pareto front and the Pareto optimal solutions obtained with the modified NSGA-II for this problem.

Figure B.3: PF and Pareto optimal solutions obtained -Example B.1-Case-2

************