Derek Corniello
EECE6046 HW3 Writeup

Task 1:

For the SGA, I studied population size, mutation rate, and crossover rate. For population, I expect the largest population to provide the best results given what we have learned in class. The larger the population, the more “chances” you have to find the best fit in a generation. This would theoretically give us a quicker convergence and a quicker discovery of the minimum. The mutation rate and crossover rate should be tweaked in order to figure out how to balance exploration and exploitation of the problem.

For the ES, I studied mu and lambda, the child selection process (mu, lambda or mu, mu+lambda), and the mutation strength. similarly to the SGA, I think that the larger population in mu and lambda would have increased the number of samples in order to narrow it down to the minimum. I originally used mu + lambda for my original assignment, as it would provide a larger overall population to sample from in parent selection. The mutation strength also allowed for us to have more of a control over the exploration vs exploitation.

Prior to the experiments, I would assume that my algorithms would perform well on the 2 dimensional space, as we have done that in a previous assignment. However, as dimensionality goes up, I expect that the minima that is found will be less accurate to the actual answer.

Task 2:

I used a simple “best fitness” approach. After a certain number of generations, I took the best minimum fitness achieved and used that as my utility measure. This is very easy to compute and compare across different runs and is probably the most clear indication of the effectiveness of the algorithm. I think that this approach is good for many other EAs, as all algorithms will have a measure of fitness no matter their setup. This can also be set up to get an average best fitness to ensure the data is not one-off.

Task 3A:

The experimental plan involved doing parameter sweeps for both SGA and ES algorithms. The SGA algorithm used 16 bits per dimension, mapped to the range given of -5.12 to 5.11. For that SGA, I used 4 population sizes, 4 mutation rates, and 4 crossover rates, resulting in 64 configurations, while ES uses 3 mu values, 3 lambda values, and 4 sigma values for 36 configurations. Each configuration is run 5 times with different seeds for statistical significance, totaling 320 SGA and 180 ES experiments. Both the algorithms were then further tested and refined with 2, 3, 5, 10, and 20 dimensionalities. Data processing averages fitness across runs, analyzes parameter effects statistically, compares performance by dimensionality, and identifies optimal regions.

Task 3B:

Based on the experimental data, the best parameter settings for the SGA are a population size of 2000, mutation rate of 0.2, crossover rate of 0.5, tournament selection with a size of 5, and elitism, with adaptive scaling for mutation and crossover rates based on problem dimensionality. For the ES, the optimal configuration is mu value of 15, lambda value of 150, and sigma at 0.1, utilizing self-adaptive mutation strengths. When comparing the two algorithms head-to-head on DeJong's F2 function using the best fitness achieved after fixed generations as the utility measure, ES with self-adaptive mutation performs dramatically better overall, achieving about the same in the low dimensions and superior scaling in higher dimensions, while SGA with tournament selection and elitism remains competitive only in medium dimensions like 5D but cannot match ES's advanced adaptation capabilities. Supporting data shows ES outperforms SGA across all runs on average, and ES exhibits more graceful performance degradation as dimensionality increases.

SGA Performance

Dimension	Best Fitness (Average)	Performance
2D	2.38e-07	Excellent
3D	0.00627	Very good
5D	0.7946	Good
10D	6.094	Acceptable
20D	44.04	Reasonable

ES Performance

Dimension	Best Fitness (Average)	Performance
2D	4.22e-24	Machine precision
3D	3.14e-05	Excellent
5D	0.5969	Very Good
10D	0.6580	Good
20D	17.87	Good

Head-to-Head Comparison

Dimension	SGA Fitness	ES Fitness	Winner	Improvement
2D	2.38e-07	4.22e-24	ES	5.6 million times better
3D	0.00627	3.14e-05	ES	200 times better
5D	0.7946	0.5969	ES	25% better
10D	6.094	0.6580	ES	9.3 times better
20D	44.04	17.87	ES	2.5 times better

Task 3C:

The experiments revealed lots of surprises beyond myinitial expectations, as ES with self-adaptive mutation achieved great precision in low dimensions and dramatically outperformed SGA across most dimensionalities, with the effectiveness of self-adaptation exceeding all predictions while SGA's improvements, though significant, were insufficient to compete with advanced ES techniques in my implementation. If repeating this assignment, I would run more parameter combinations within the tested ranges, include additional selection methods for SGA like rank-based selection, test intermediate mutation rates more thoroughly, extend runs to higher generations for convergence analysis, and perform sensitivity analysis on parameter interactions to better understand trade-offs. I believe that while I came to this conclusion with some changes, much more focused tuning could get these algorithms closer to the minima, and may be able to get them closer in performance