Differences between simple heuristic algorithms (Hill Climbing, Simulated Annealing) and Genetic Algorithms in finding Global Minimum of numerical benchmark functions

George Buțco

December 2, 2021

Abstract

This paper compares Hill Climbing and Simulated Annealing with Genetic Algorithms (Truncation Selection and Roulette Wheel Selection with k-point crossover, where k $\in 1, 2, 3, 4$) focusing on the returned value and ignoring the factor time.

The comparison will be done using De Jong 1 function[DJ75], Schwefel's function[JY13], Rastrigin's function[Ras74], Michalewicz's function[BFM97] in 5, 10, and 30 dimensions.

The quantification is given by the mean value. If the mean values are almost equal, we choose the algorithm with the lowest margin of error.

1 Introduction

"**Genetic algorithms** (GAs) were invented by John Holland in the 1960s [...], Holland's original goal was not to design algorithms to solve specific problems, but rather to formally study the phenomenon of adaptation as it occurs in nature and to develop ways in which the mechanisms of natural adaptation might be imported into computer systems. Holland's 1975 book *Adaptation in Natural and Artificial Systems*[Sam76] presented the genetic algorithm as an abstraction of biological evolution and gave a theoretical framework for adaptation under the GA.
Holland's GA is a method for moving from one population of "chromosomes" (e.g., strings of ones and zeros, or "bits") to a new population by using a kind of "natural selection" together with the genetics-inspired operators of crossover, mutation, and inversion.
Each **chromosome** consists of "genes" (e.g., bits), each **gene** being an instance of a particular "allele" (e.g., 0 or 1).
The **selection** operator chooses those chromosomes in the population that will be allowed to reproduce, and on average the fitter chromosomes produce more offspring than the less fit ones.
Crossover exchanges sub-parts of two chromosomes, roughly mimicking biological recombination between two single-chromosome [...]."[Mit98]

Algorithm 1 Genetic Algorithm

```
t ← 0
generate the starting population P_t
while not stopping condition do
    if N is even then
        evaluate(P_t)
        t ← t + 1
        P_t ← select(P_{t−1})
        P_t ← crossOver(P_t)
        P_t ← mutate(P_t)
    end if
end while
```

1.1 Truncation Selection

Truncation Selection is less sophisticated than many other selection methods, and is not often used in practice. It selects the best n chromosomes from the population.

1.2 Roulette Wheel Selection

RWS selects each solution in the population occupies an area on the roulette wheel proportional to its fitness and the roulette wheel is spun as many times as the population size. Since the solutions are marked proportionally to their fitness, a solution with a higher fitness is likely to receive more copies than a solution with a low fitness.[BFM18]

Algorithm 2 Roulette Wheel Selection

function RWS(population P)
  $n \leftarrow = \text{size}(P)$
  $E_i \leftarrow = \text{evaluate}(P_i)$
  $E_{max} \leftarrow = \max(E)$
  $E_{min} \leftarrow = \text{in}(E)$
  $F_i \leftarrow \frac{E_{max}-E_i}{E_{max}-E_{min}+000001} + 0.01$
  $F_{sum} \leftarrow \sum_{i=1}^{n} F_i$
  $Fn_i \leftarrow \frac{F_i}{F_{sum}}$
  $pc_1 \leftarrow Fn_1$
  for $i \leftarrow 2$ to n **do**
    $pc_i \leftarrow Fn_i + pc_{i-1}$
  end for
  $pc_n \leftarrow 1$

  for $i \leftarrow 1$ to n **do**
    $r \leftarrow \text{random}[01)$
    for $j \leftarrow 1$ to n **do**
      if $r \leq pc_j$ **then**
        push P_j in P_{next}
        break
      end if
    end for
  end for
  return P_{next}
end function

1.3 k-point crossover

In k-point crossover, k crossover points are picked from the parent chromosomes. The bits in between the k points are swapped between the parent organisms.

1.4 Function Definition

1.4.1 De Jong 1 Function

$$f(x) = \sum_{i=1}^{n} x_i^2 \qquad -5.12 \leq x_i \leq 5.12$$
$$min(f(x)) = 0$$

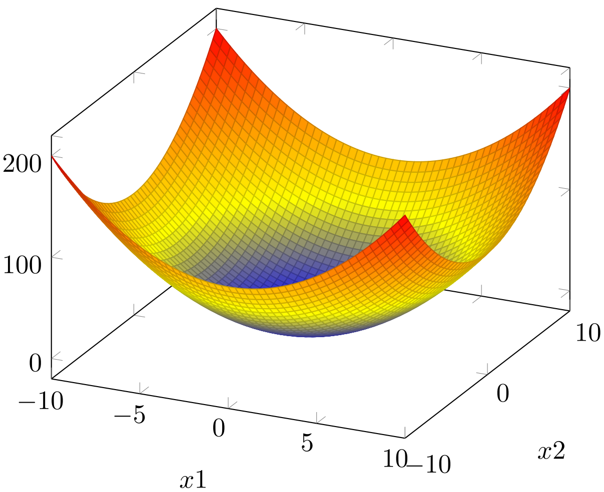


Figure 1: DeJong's Function Graphic

1.4.2 Schwefel's Function

$$f(x) = \sum_{i=1}^{n} (-x_i sin(\sqrt{|x_i|})) \qquad -500 \leq x_i \leq 500$$
$$min(f(x)) = -n \cdot 418.9829$$

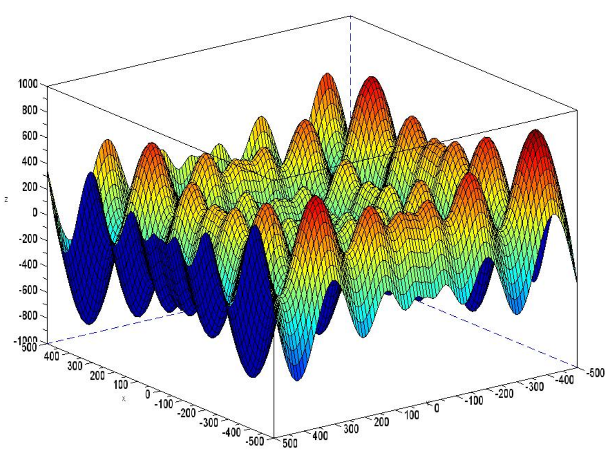


Figure 2: Schwefel's Function Graphic[McC]

1.4.3 Michalewicz's Function

$$f(x) = -\sum_{i=1}^{n} sin(x_i) \cdot \left(sin \left(\frac{i \cdot x_i^2}{\pi} \right) \right)^{2 \cdot m} \qquad m = 10, \quad 0 \leq x_i \leq \pi$$

$min(f(x)) = -4.6876 \qquad n = 5$

$min(f(x)) = -9.6601 \qquad n = 10$

$min(f(x)) = -29.631 \qquad n = 30$

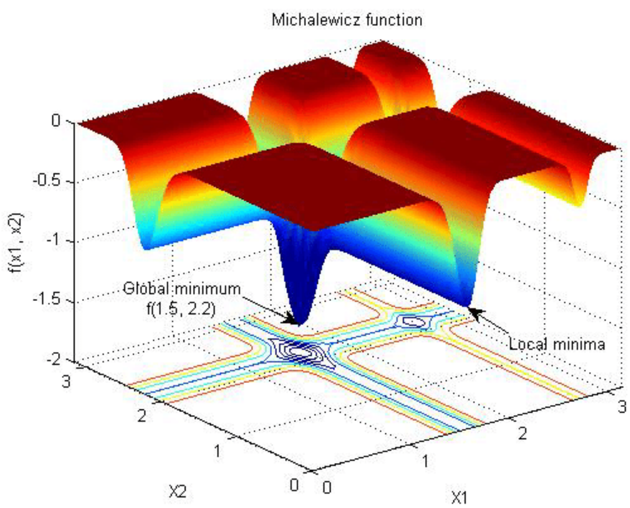


Figure 3: Michalewicz's Function Graphic[Moo]

1.4.4 Rastrigin's Function

$$f(x) = 10 \cdot n + \sum_{i=1}^{n} (x_i^2 - 10 \cdot cos(2 \cdot \pi \cdot x_i)) \qquad -5.12 \leq x_i \leq 5.12$$

$min(f(x)) = 0$

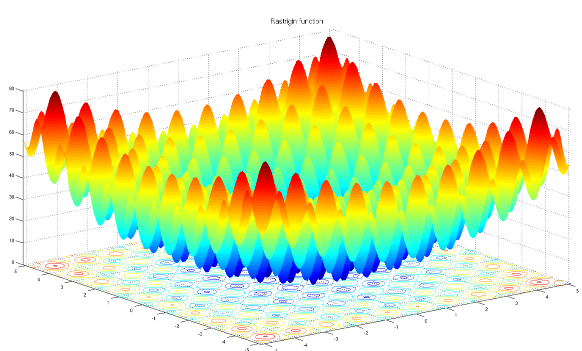


Figure 4: Rastrigin's Function Graphic[Wik]

1.5 Confidence Interval

The confidence interval[Bal96] is defined as follows:

$$CI = \overline{x} \pm z\frac{\sigma}{\sqrt{n}}$$

$\overline{x}$ is the sample **mean**
z is the **confidence level**. 1.645 (90%) is the convention standard
σ is the sample **standard deviation**
n is the sample **size**

1.6 Describe Comparison Method

Every function (De Jong 1 function, Schwefel's function, Rastrigin's function, Michalewicz's function) will be tested for 5, 10, and 30 dimensions.

The comparison is made between different versions of Genetic Algorithms(for selection is used Truncation Selection or Roulette Wheel Selection, and for crossover, k-point crossover where $k \in 1, 2, 3, 4$) on every benchmark function version.
The number of generations for every Ga is 1000, and the sample size for each test is 30.

The best Genetic Algorithm for each benchmark function is compared with the best strategy between Hill Climbing and Simulated Annealing for the same benchmark function [Bu1].
The best strategy has the lowest mean and, if the mean is similar (the absolute value of the difference is lower than 0.1), is chosen the one with a lower margin of error.

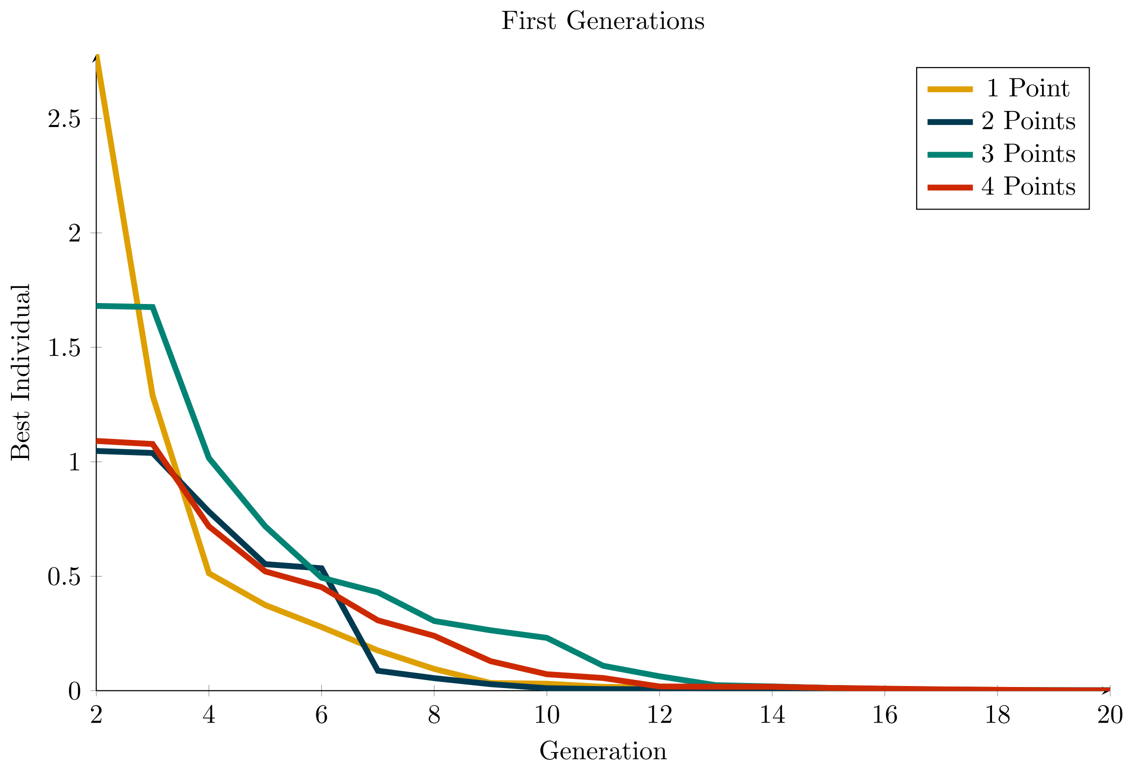
1.6.1 Hypotheses

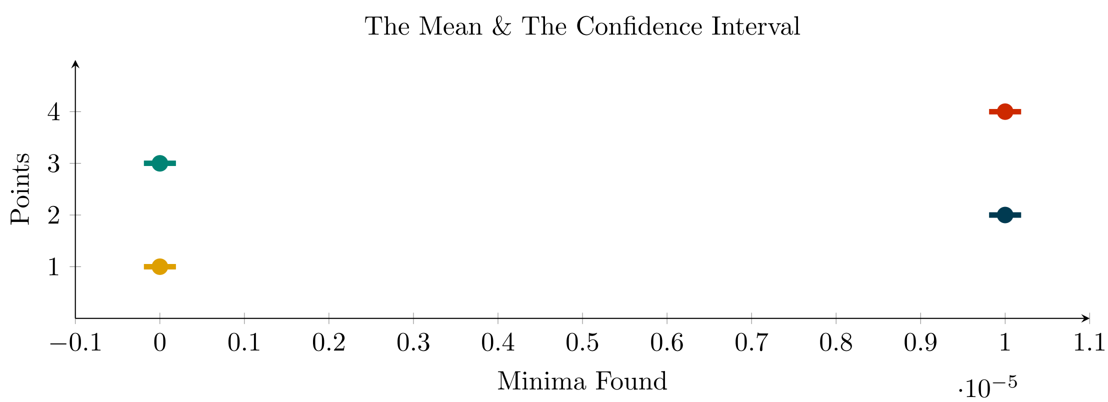
1. RWS works better compared to Truncation Selection(TS), at least for wave functions
2. Higher dimensions need more points in the k-point crossover
3. The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner
4. HAs work better for paraboloid types of functions due to their Hill Climbing Nature

2 Genetic Algorithm Comparison

2.1 De Jong 1 Function

2.1.1 5 dimensions - Truncate Selection

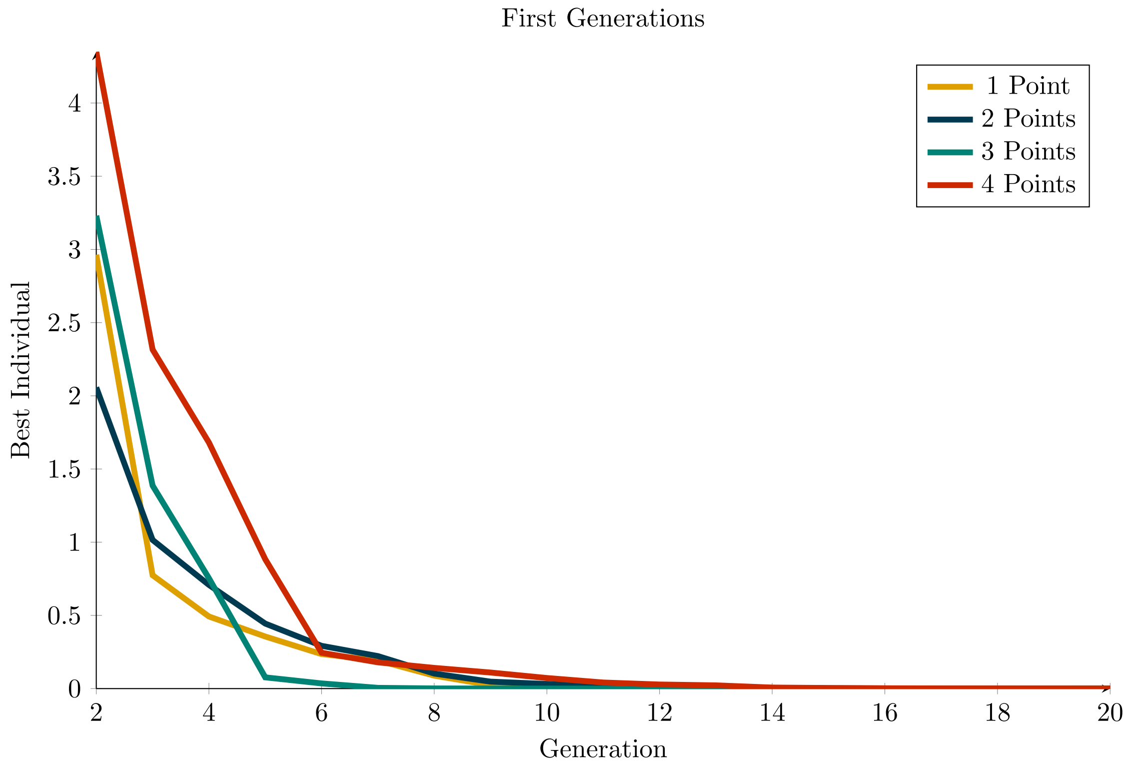


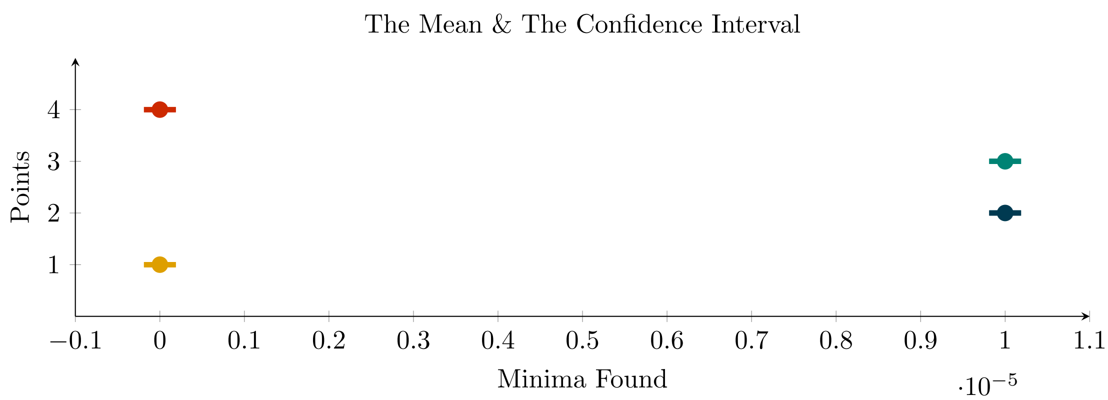


	Mean	MOE	Best	Worst
1 Point	0.00000	0.00000	0.00000	0.00001
2 Points	0.00001	0.00000	0.00000	0.00001
3 Points	0.00000	0.00000	0.00000	0.00001
4 Points	0.00001	0.00000	0.00000	0.00001

Figure 5: De Jong 1 Function - 5 dimensions - Truncate Selection

2.1.2 5 dimensions - Roulette Wheel Selection

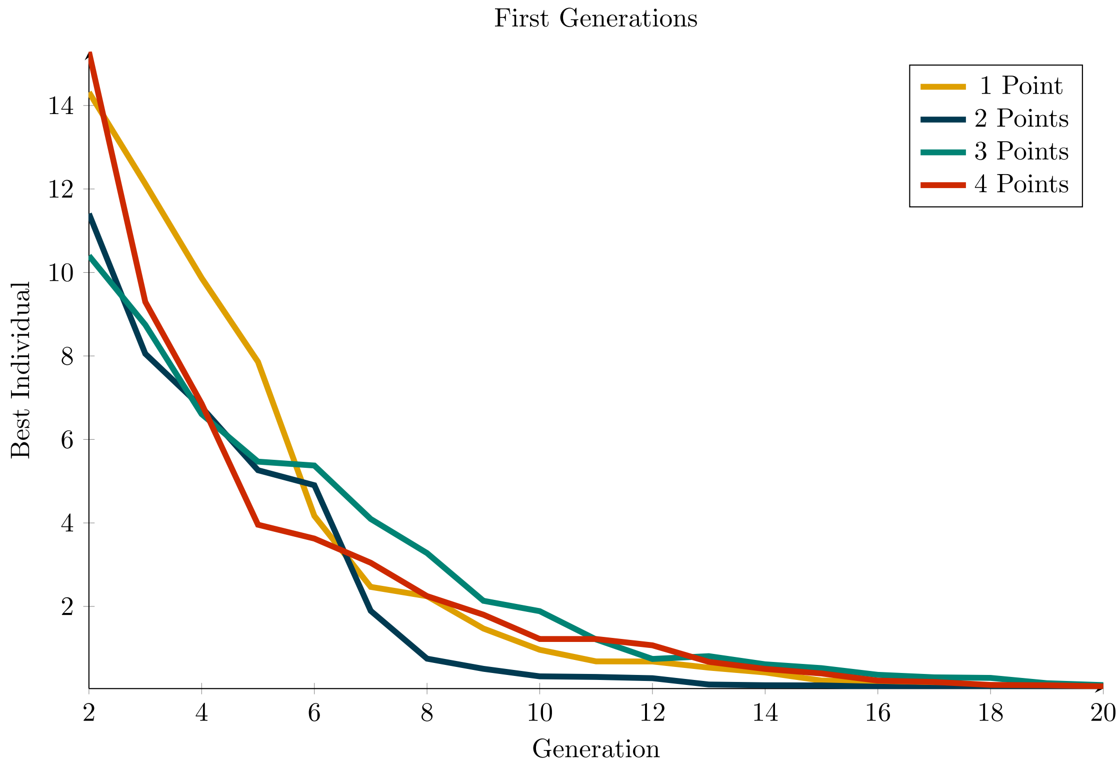




	Mean	MOE	Best	Worst
1 Point	0.00000	0.00000	0.00000	0.00001
2 Points	0.00001	0.00000	0.00000	0.00001
3 Points	0.00001	0.00000	0.00000	0.00001
4 Points	0.00000	0.00000	0.00000	0.00001

Figure 6: De Jong 1 Function - 5 dimensions - Roulette Wheel Selection

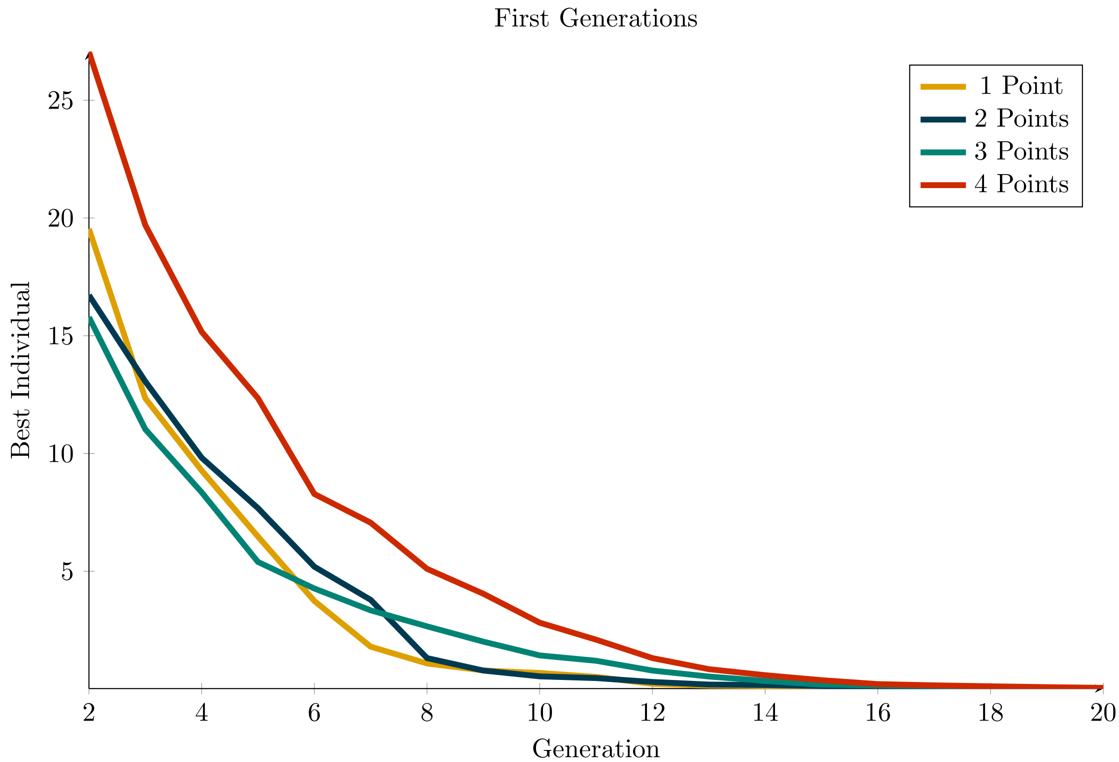
2.1.3 10 dimensions - Truncate Selection

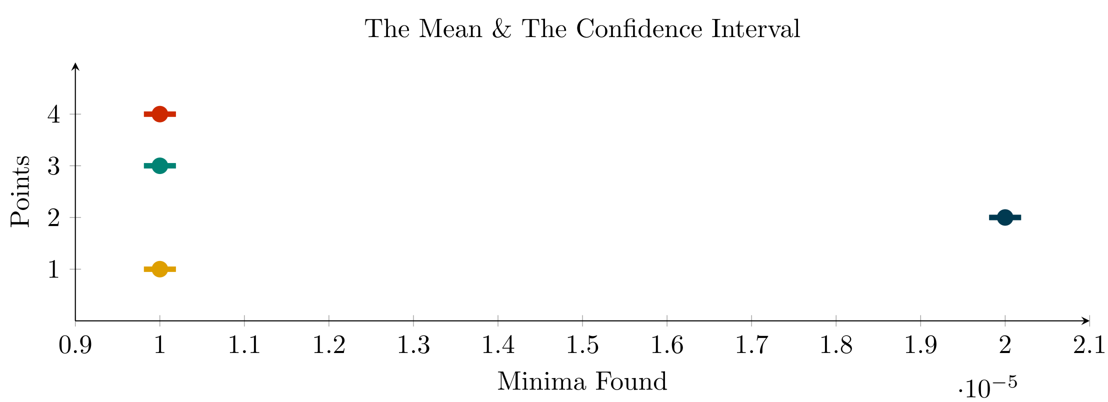


	Mean	MOE	Best	Worst
1 Point	0.00001	0.00000	0.00000	0.00001
2 Points	0.00000	0.00000	0.00000	0.00001
3 Points	0.00001	0.00000	0.00000	0.00001
4 Points	0.00001	0.00000	0.00000	0.00002

Figure 7: De Jong 1 Function - 10 dimensions - Truncate Selection

2.1.4 10 dimensions - Roulette Wheel Selection

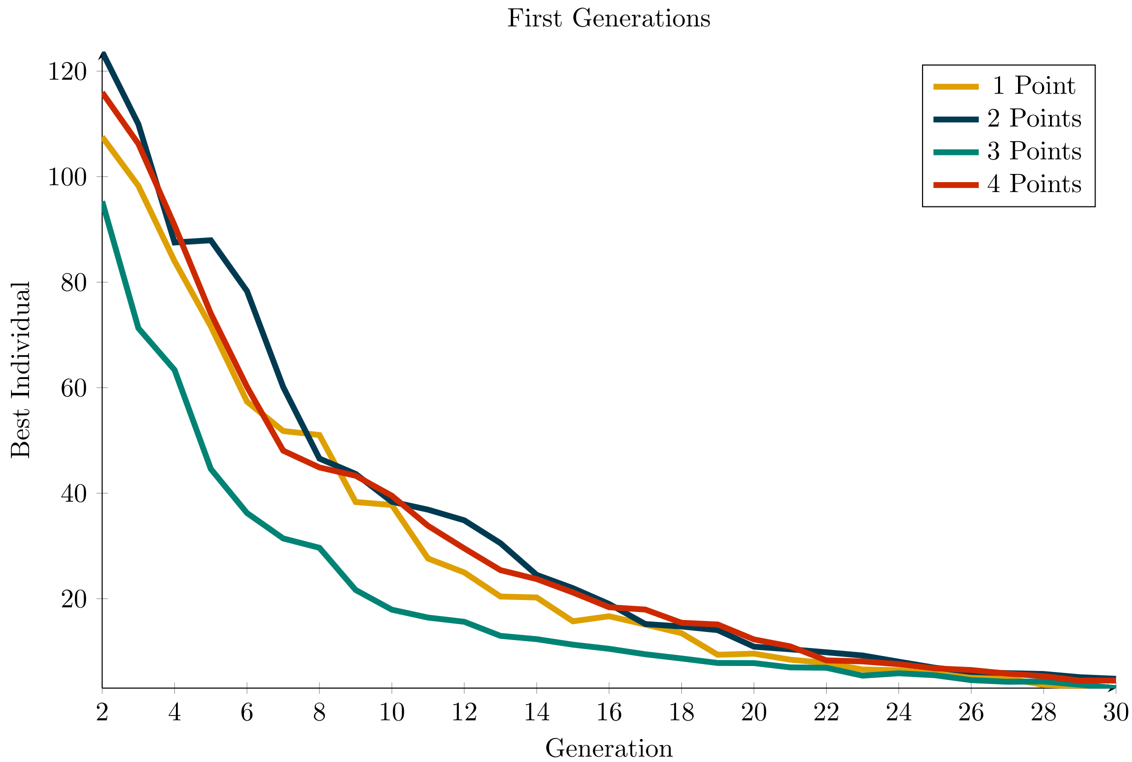


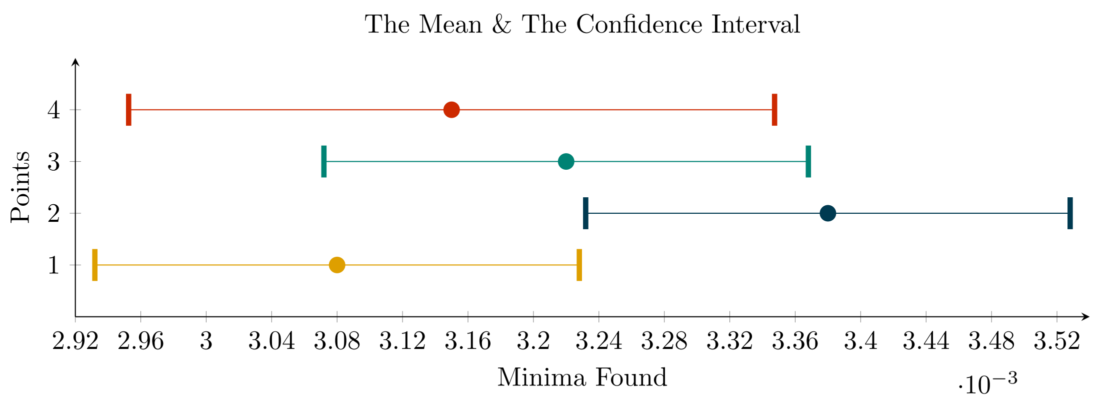


	Mean	MOE	Best	Worst
1 Point	0.00001	0.00000	0.00000	0.00001
2 Points	0.00002	0.00000	0.00001	0.00003
3 Points	0.00001	0.00000	0.00000	0.00001
4 Points	0.00001	0.00000	0.00000	0.00001

Figure 8: De Jong 1 Function - 10 dimensions - Roulette Wheel Selection

2.1.5 30 dimensions - Truncate Selection

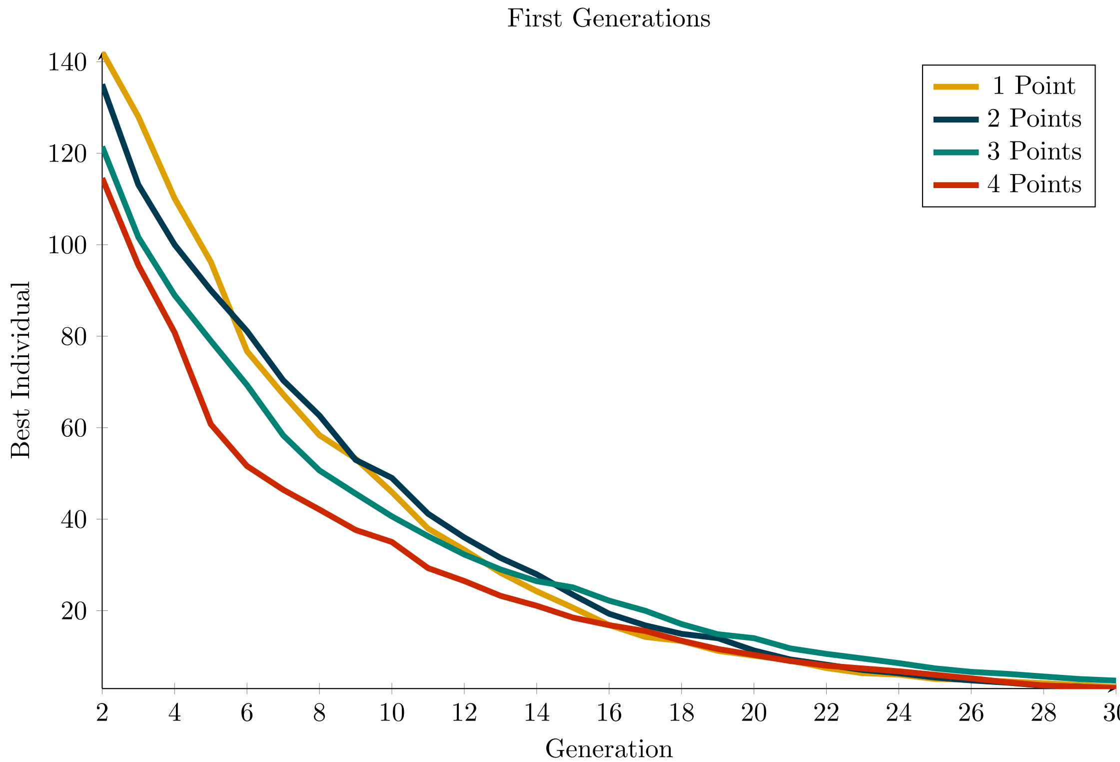


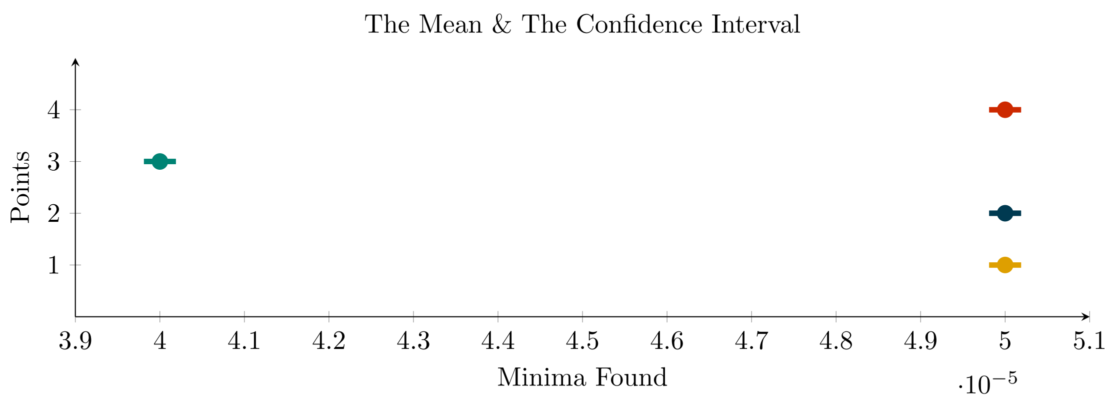


	Mean	MOE	Best	Worst
1 Point	0.00308	0.00009	0.00185	0.00399
2 Points	0.00338	0.00009	0.00240	0.00423
3 Points	0.00322	0.00009	0.00227	0.00411
4 Points	0.00315	0.00012	0.00169	0.00509

Figure 9: De Jong 1 Function - 30 dimensions - Truncate Selection

2.1.6 30 dimensions - Roulette Wheel Selection



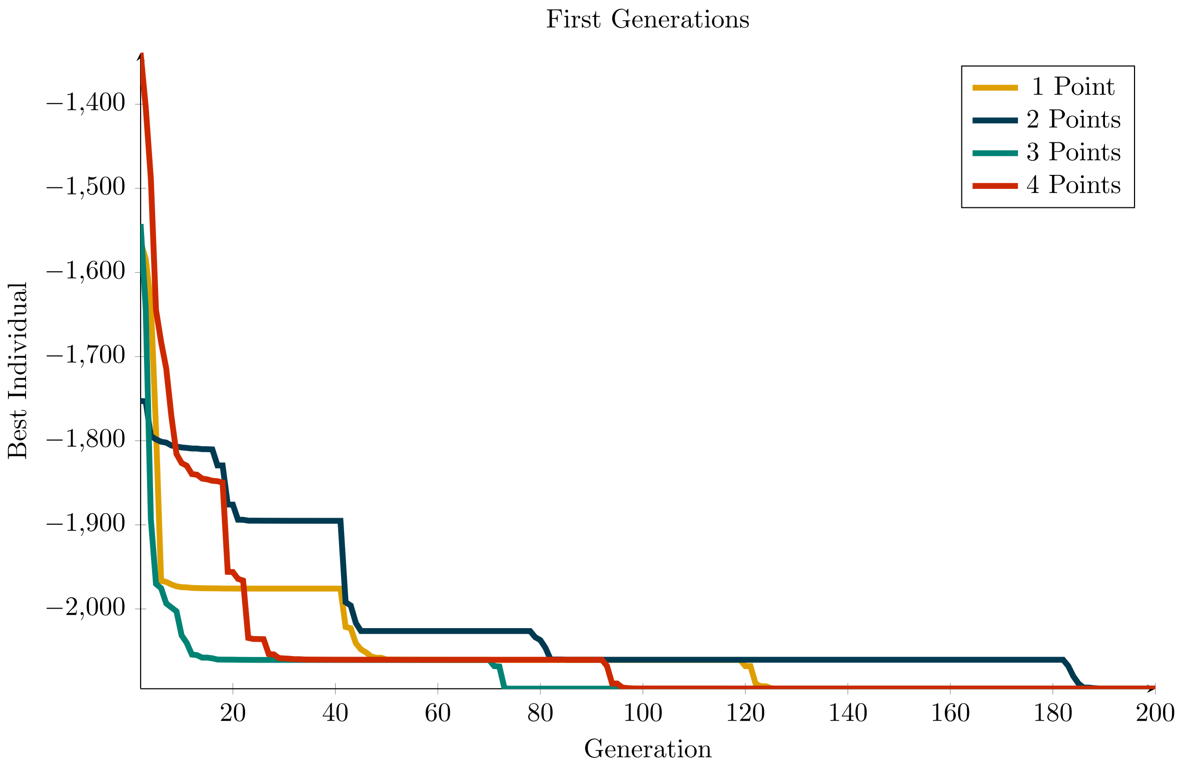


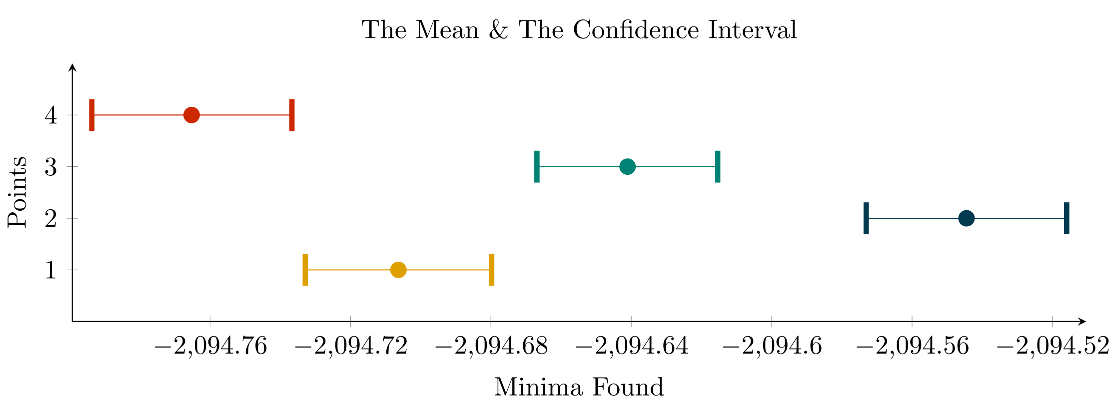
	Mean	MOE	Best	Worst
1 Point	0.00005	0.00000	0.00004	0.00006
2 Points	0.00005	0.00000	0.00004	0.00006
3 Points	0.00004	0.00000	0.00003	0.00005
4 Points	0.00005	0.00000	0.00004	0.00007

Figure 10: De Jong 1 Function - 30 dimensions - Roulette Wheel Selection

2.2 Schwefel's Function

2.2.1 5 dimensions - Truncate Selection

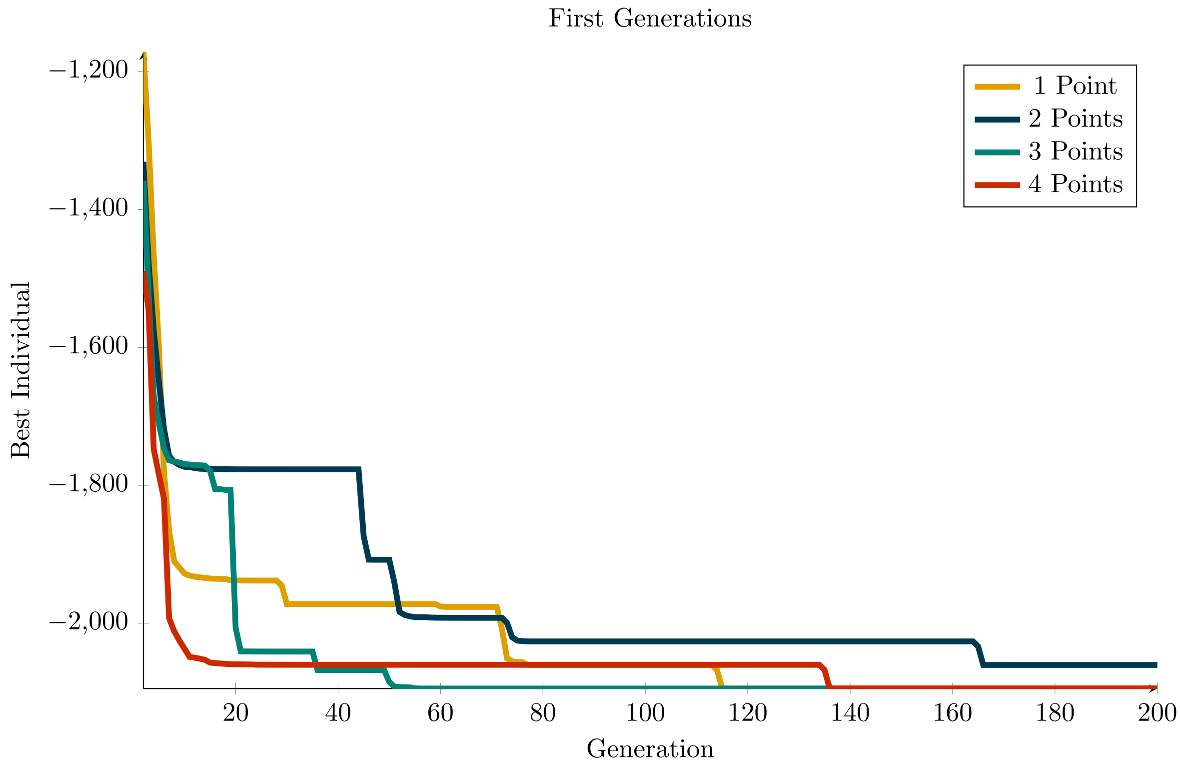


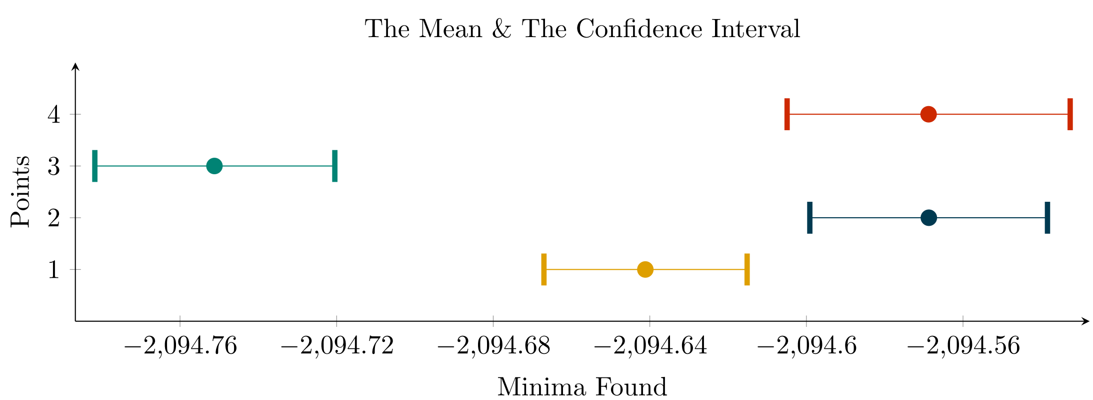


	Mean	MOE	Best	Worst
1 Point	-2094.70615	0.01616	-2094.91318	-2094.49914
2 Points	-2094.54433	0.01738	-2094.70709	-2094.39610
3 Points	-2094.64089	0.01576	-2094.91252	-2094.49975
4 Points	-2094.76508	0.01736	-2094.91378	-2094.60281

Figure 11: Schwefel's Function - 5 dimensions - Truncation Selection

2.2.2 5 dimensions - Roulette Wheel Selection

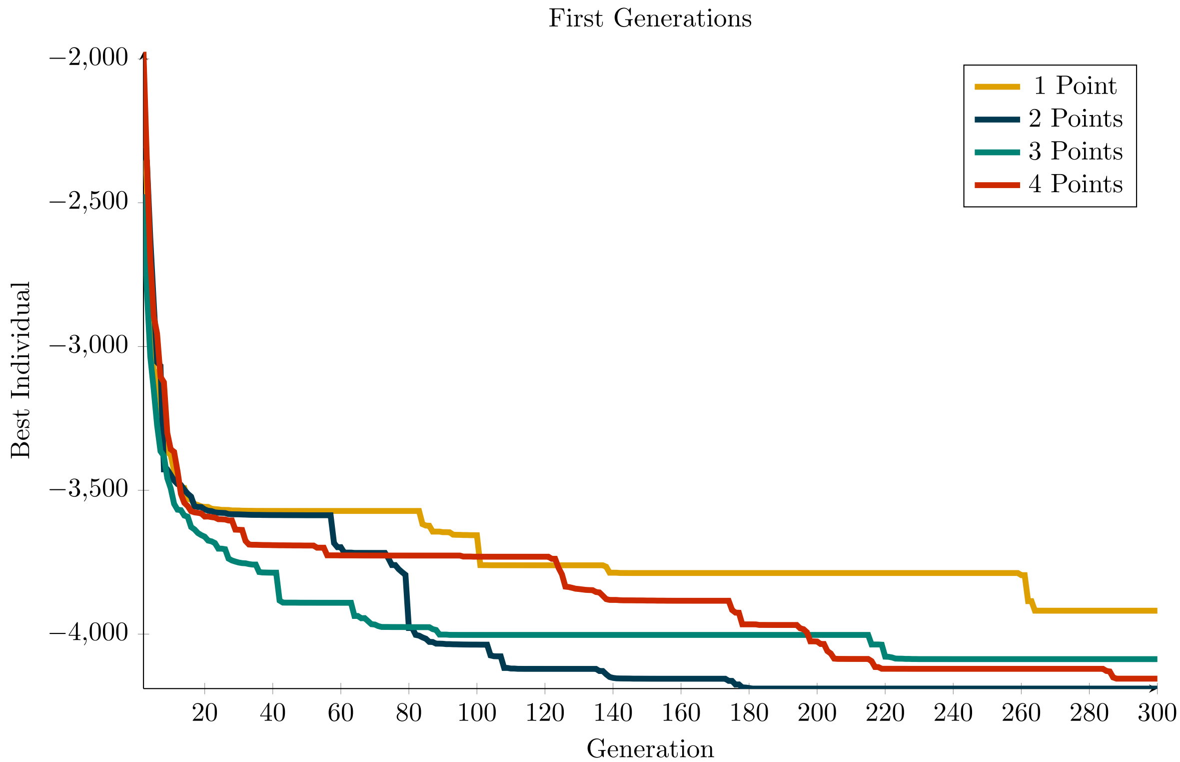


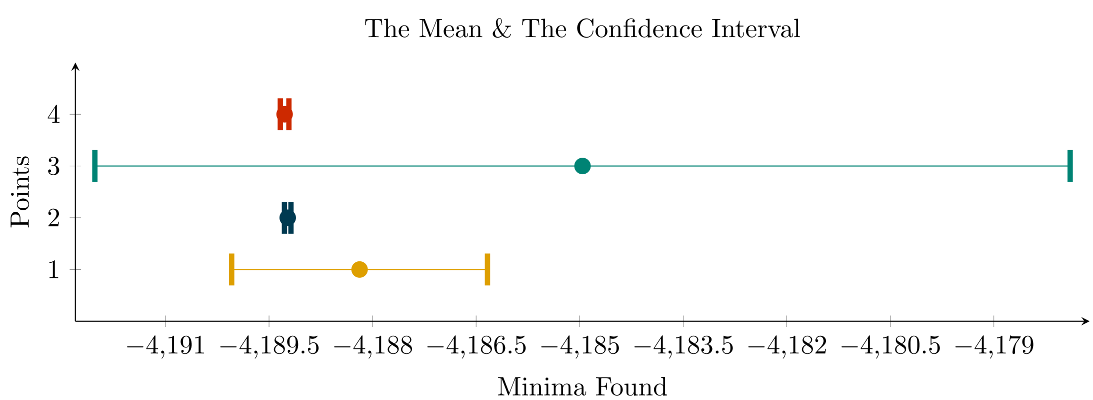


	Mean	MOE	Best	Worst
1 Point	-2094.64100	0.01578	-2094.81014	-2094.49975
2 Points	-2094.56860	0.01847	-2094.81014	-2094.39610
3 Points	-2094.75105	0.01871	-2094.91379	-2094.49914
4 Points	-2094.56866	0.02201	-2094.81075	-2094.39610

Figure 12: Schwefel's Function - 5 dimensions - Roulette Wheel Selection

2.2.3 10 dimensions - Truncate Selection

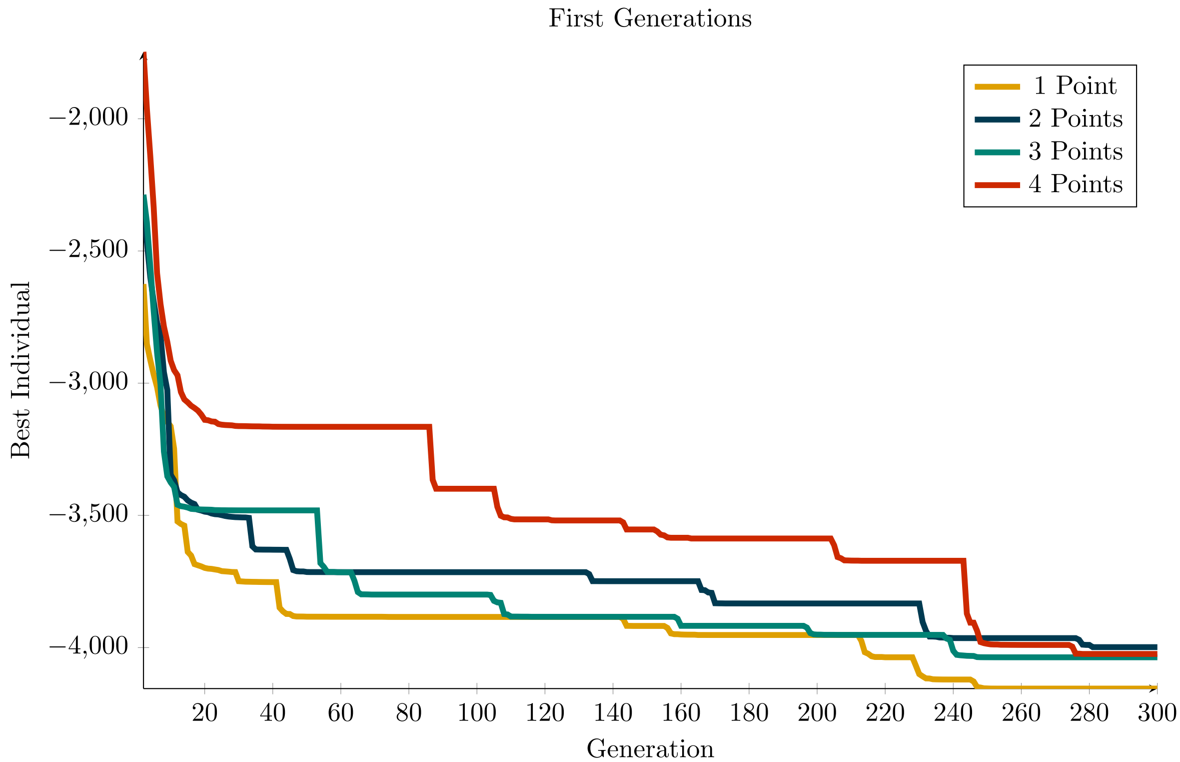


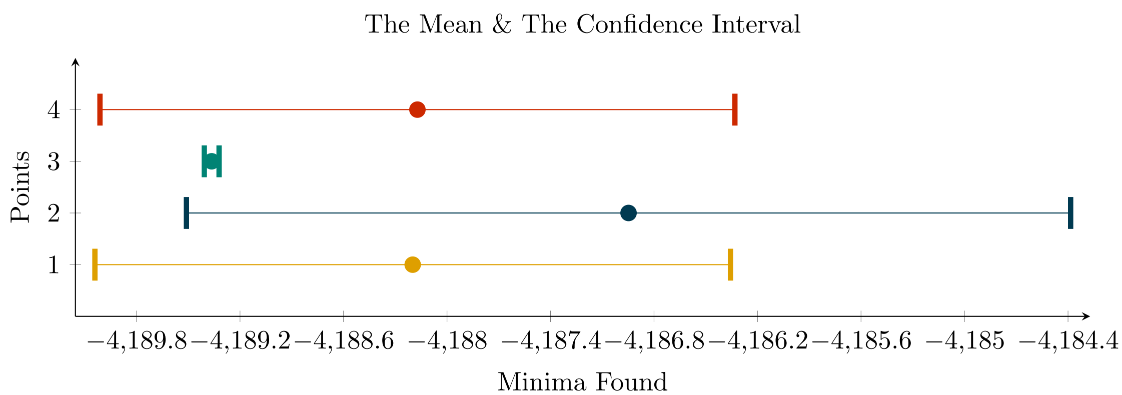


	Mean	MOE	Best	Worst
1 Point	-4188.18872	1.12644	-4189.62024	-4154.97008
2 Points	-4189.22986	0.02871	-4189.51598	-4188.89523
3 Points	-4184.95907	4.29454	-4189.61966	-4058.29043
4 Points	-4189.27495	0.03790	-4189.72331	-4188.89524

Figure 13: Schwefel's Function - 10 dimensions - Truncation Selection

2.2.4 10 dimensions - Roulette Wheel Selection

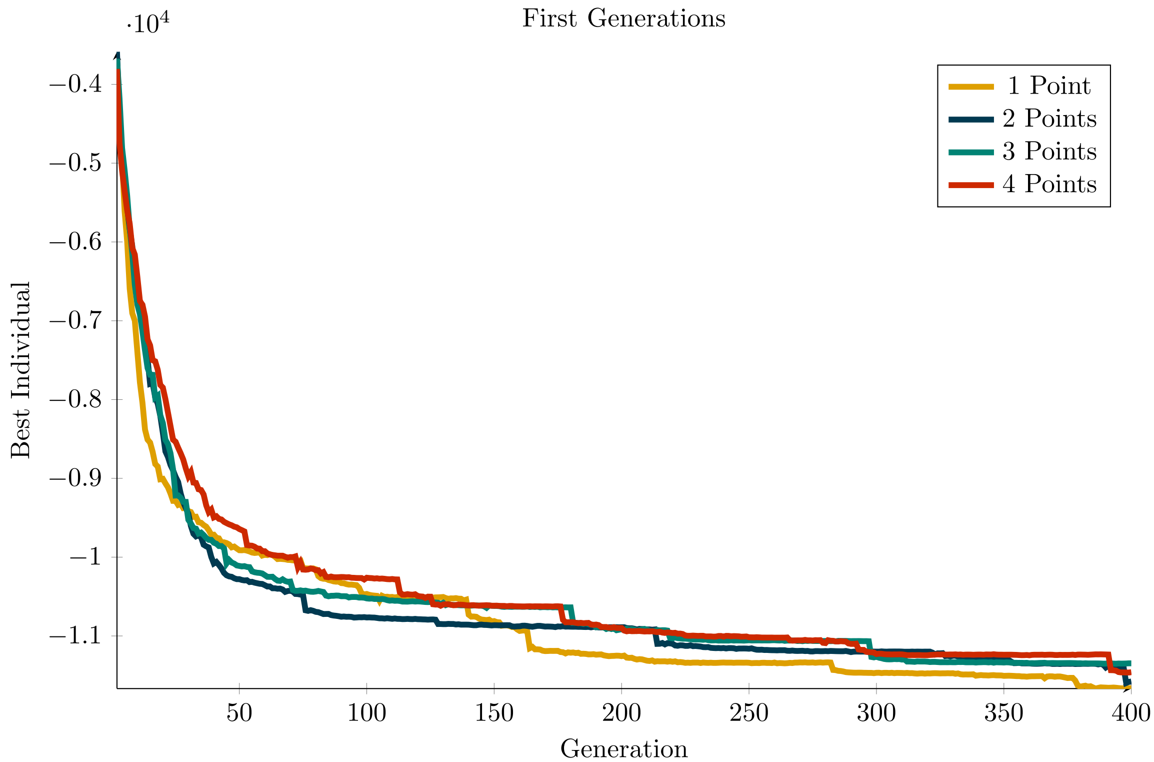


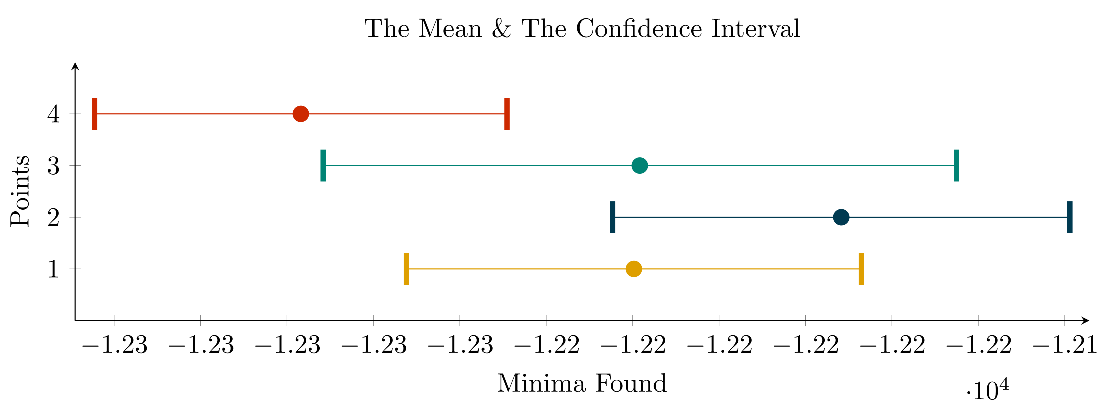


	Mean	MOE	Best	Worst
1 Point	-4188.19907	1.11991	-4189.62086	-4155.17771
2 Points	-4186.94773	1.55802	-4189.41355	-4154.86706
3 Points	-4189.36437	0.02613	-4189.72298	-4189.10254
4 Points	-4188.17160	1.11884	-4189.51659	-4155.17741

Figure 14: Schwefel's Function - 10 dimensions - Roulette Wheel Selection

2.2.5 30 dimensions - Truncate Selection

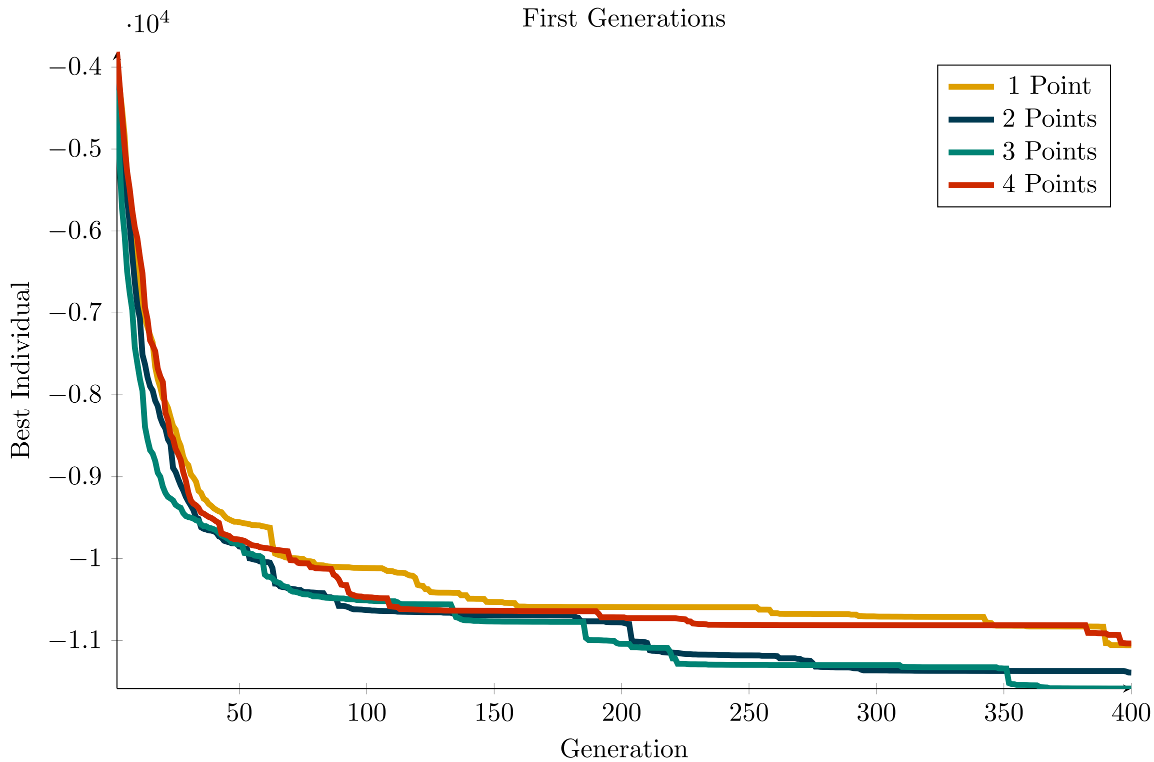


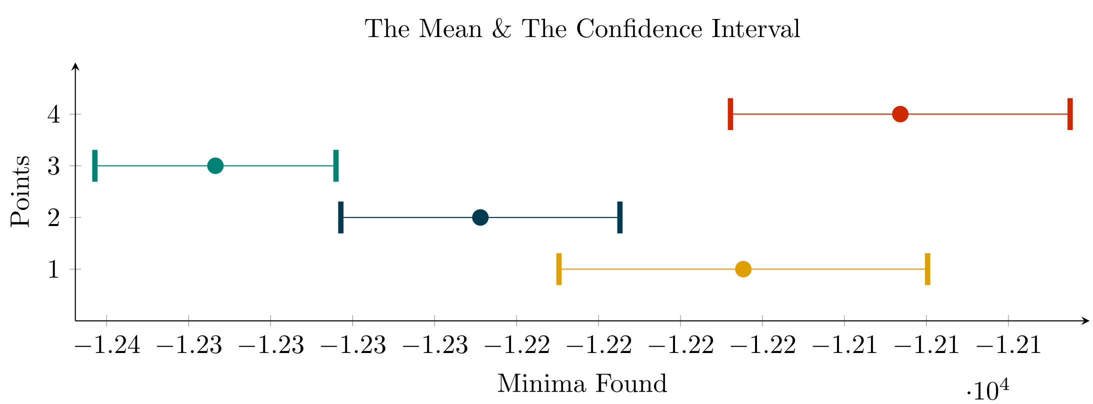


	Mean	MOE	Best	Worst
1 Point	-12223.76798	25.60556	-12524.06430	-11912.78824
2 Points	-12185.37124	25.73434	-12492.62036	-11896.50800
3 Points	-12222.67653	35.63991	-12561.04219	-11867.44762
4 Points	-12285.43489	23.20869	-12456.59797	-11933.74200

Figure 15: Schwefel's Function - 30 dimensions - Truncation Selection

2.2.6 30 dimensions - Roulette Wheel Selection



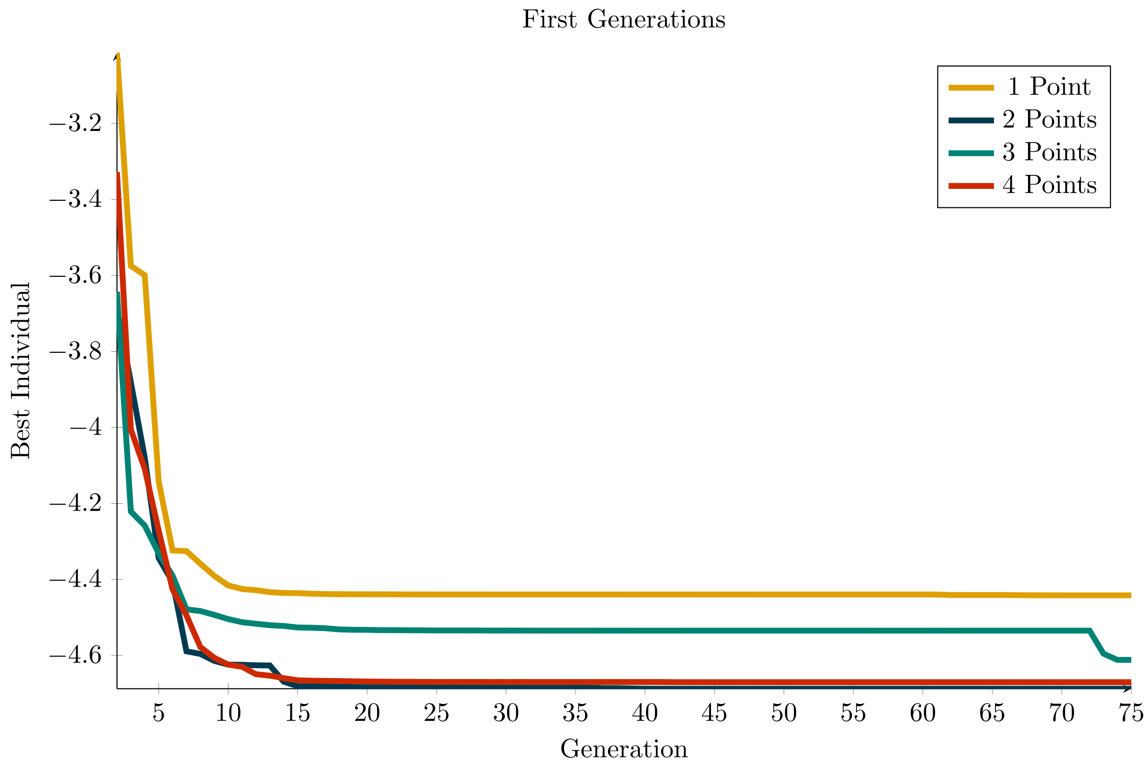


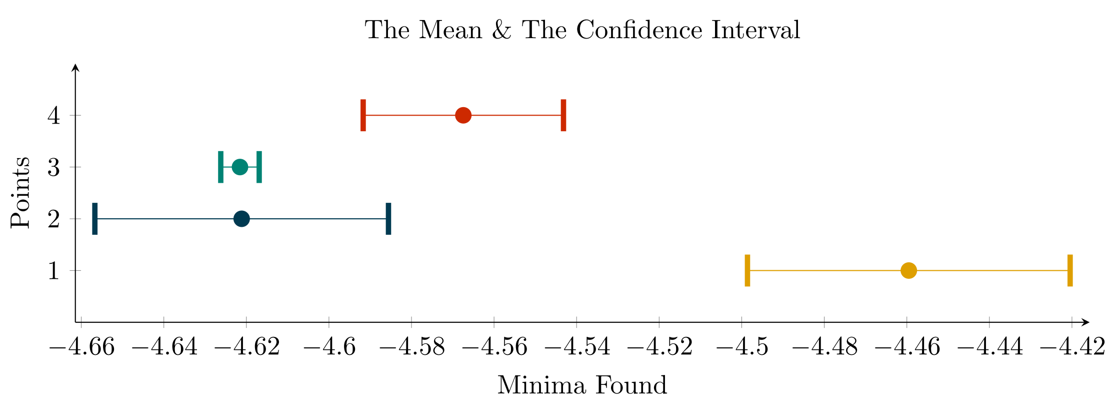
	Mean	MOE	Best	Worst
1 Point	-12156.30729	36.90360	-12499.56634	-11678.09258
2 Points	-12242.90097	27.94104	-12498.98856	-11913.41438
3 Points	-12330.16227	24.13945	-12499.82851	-11994.89017
4 Points	-12104.60284	34.00596	-12346.56254	-11634.05282

Figure 16: Schwefel's Function - 30 dimensions - Roulette Wheel Selection

2.3 Michalewicz's Function

2.3.1 5 dimensions - Truncate Selection

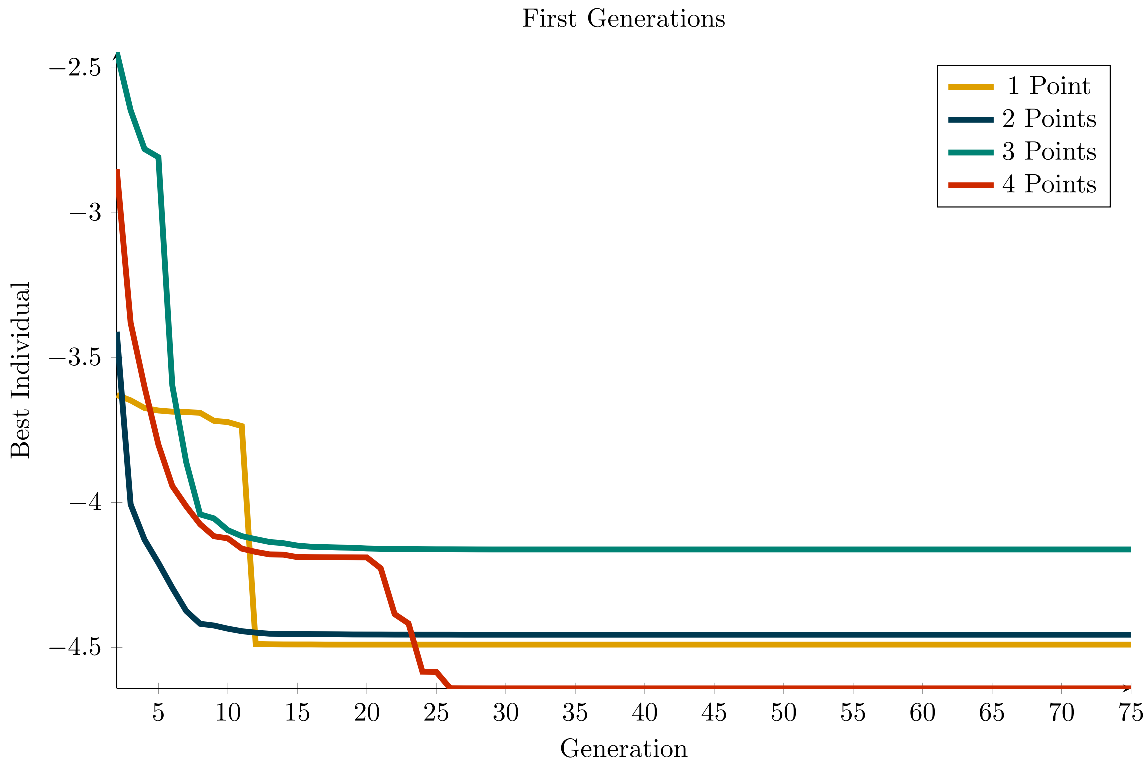


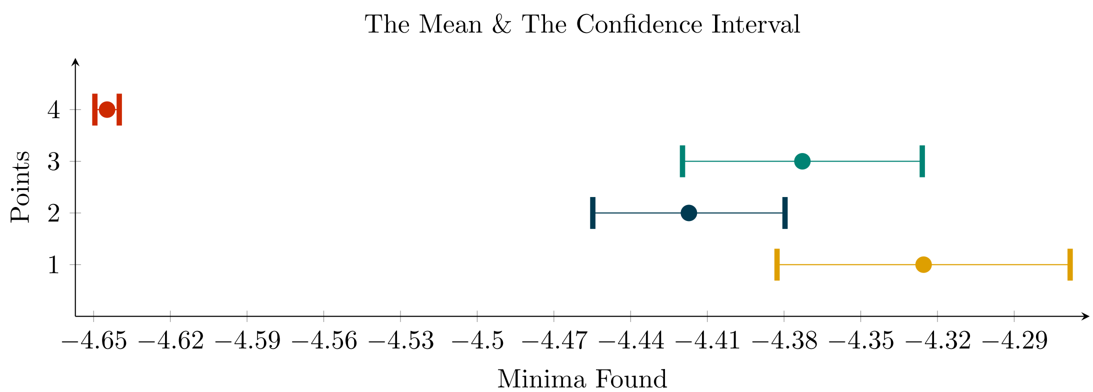


	Mean	MOE	Best	Worst
1 Point	-4.45952	0.02376	-4.68593	-4.12017
2 Points	-4.62114	0.02162	-4.68765	-4.20740
3 Points	-4.62155	0.00283	-4.65247	-4.61070
4 Points	-4.56744	0.01475	-4.68758	-4.46561

Figure 17: Michalewicz - 5 dimensions - Truncate Selection

2.3.2 5 dimensions - Roulette Wheel Selection

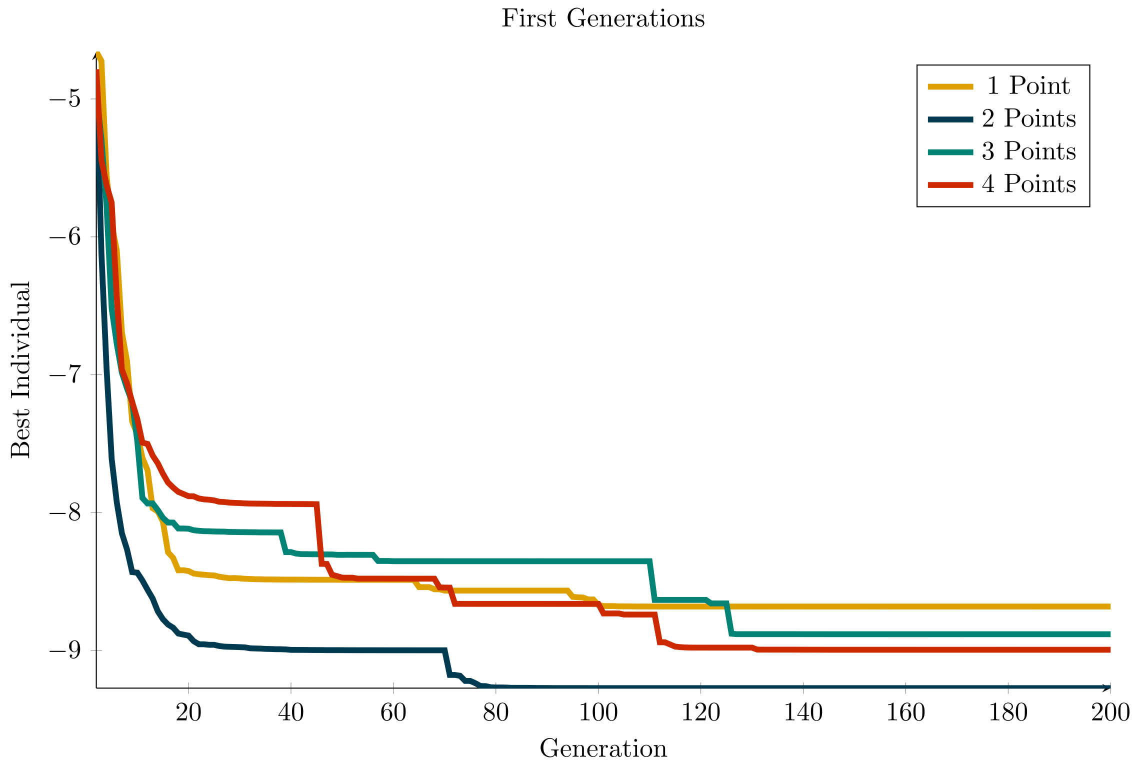


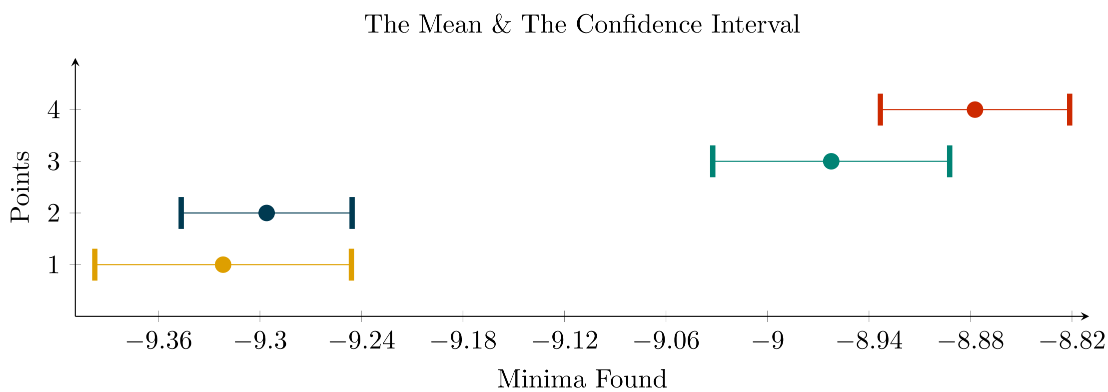


	Mean	MOE	Best	Worst
1 Point	-4.32542	0.03484	-4.68312	-4.12636
2 Points	-4.41720	0.02285	-4.53257	-4.16924
3 Points	-4.37282	0.02850	-4.52099	-4.15135
4 Points	-4.64474	0.00290	-4.68593	-4.61166

Figure 18: Michalewicz - 5 dimensions - Roulette Wheel Selection

2.3.3 10 dimensions - Truncate Selection

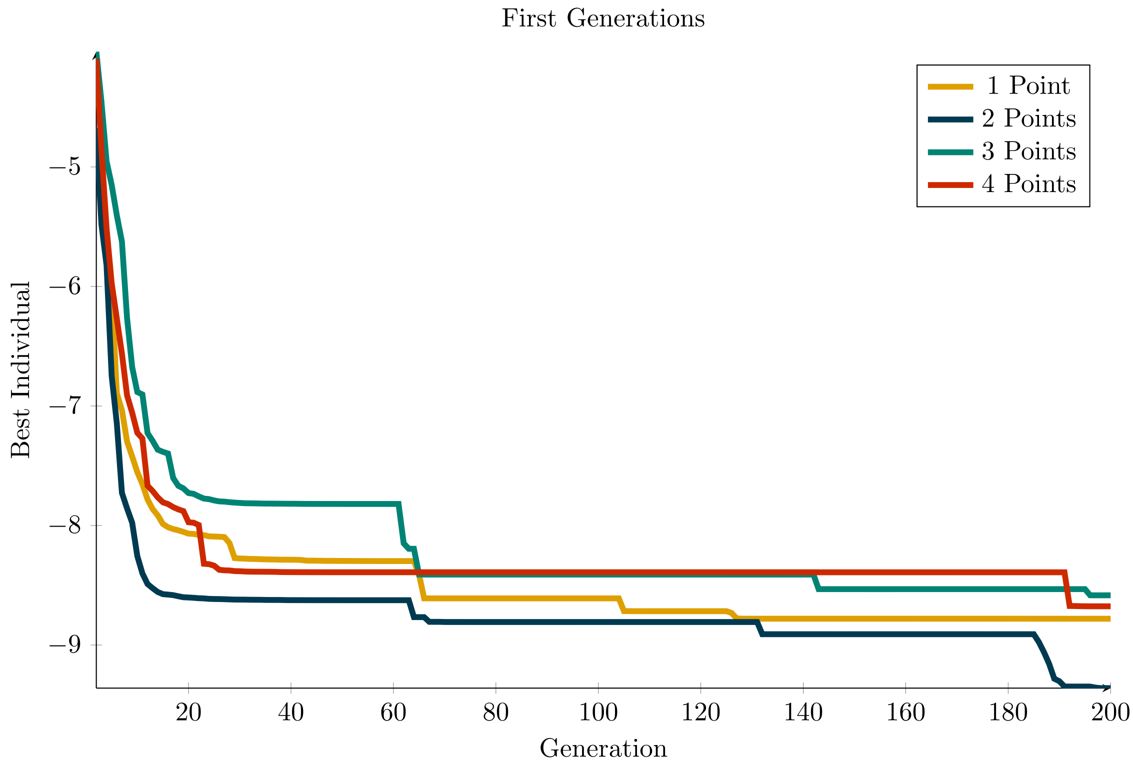


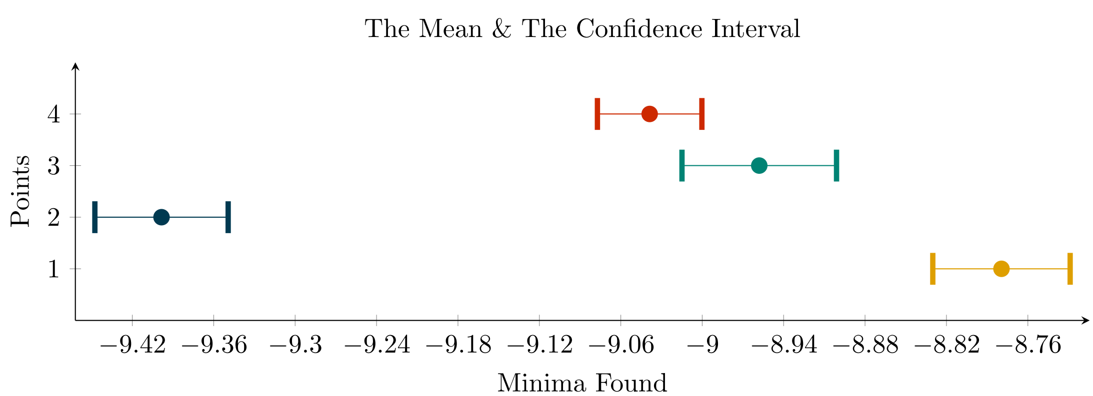


	Mean	MOE	Best	Worst
1 Point	-9.32179	0.04610	-9.65521	-8.71418
2 Points	-9.29605	0.03071	-9.56444	-8.75595
3 Points	-8.96231	0.04257	-9.32817	-8.46847
4 Points	-8.87732	0.03402	-9.36792	-8.57361

Figure 19: Michalewicz - 10 dimensions - Truncate Selection

2.3.4 10 dimensions - Roulette Wheel Selection

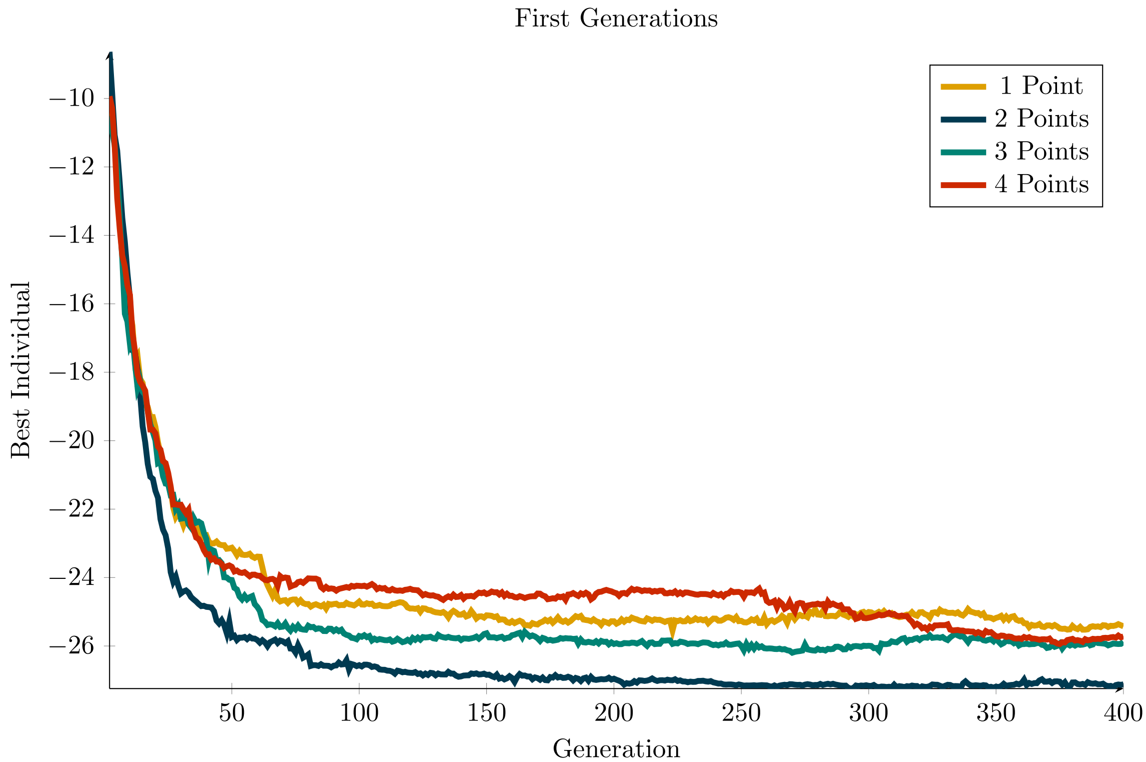


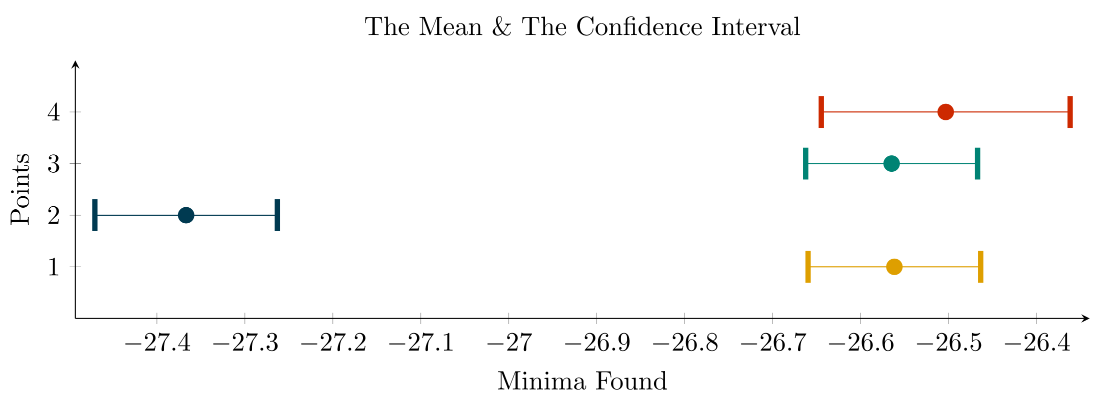


	Mean	MOE	Best	Worst
1 Point	-8.77944	0.03077	-9.27489	-8.50039
2 Points	-9.39847	0.02986	-9.56335	-8.81712
3 Points	-8.95795	0.03462	-9.24780	-8.50694
4 Points	-9.03872	0.02340	-9.26046	-8.82130

Figure 20: Michalewicz - 10 dimensions - Roulette Wheel Selection

2.3.5 30 dimensions - Truncate Selection

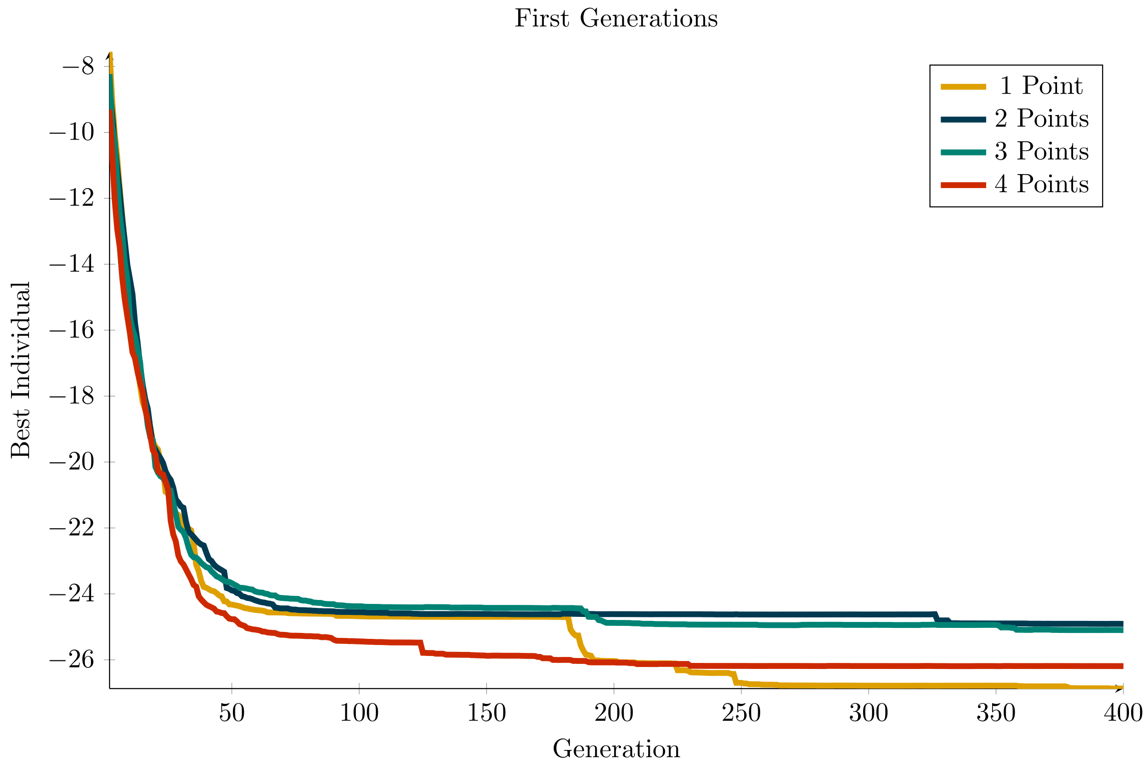


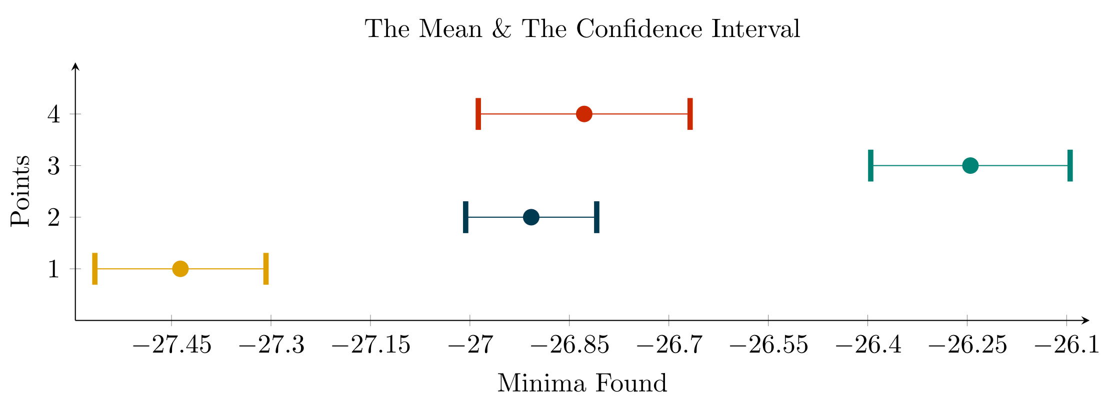


	Mean	MOE	Best	Worst
1 Point	-26.56163	0.05964	-27.40644	-26.02390
2 Points	-27.36678	0.06306	-27.92185	-26.64083
3 Points	-26.56476	0.05940	-27.32469	-25.90648
4 Points	-26.50327	0.08599	-27.47655	-25.40123

Figure 21: Michalewicz - 30 dimensions - Truncate Selection

2.3.6 30 dimensions - Roulette Wheel Selection



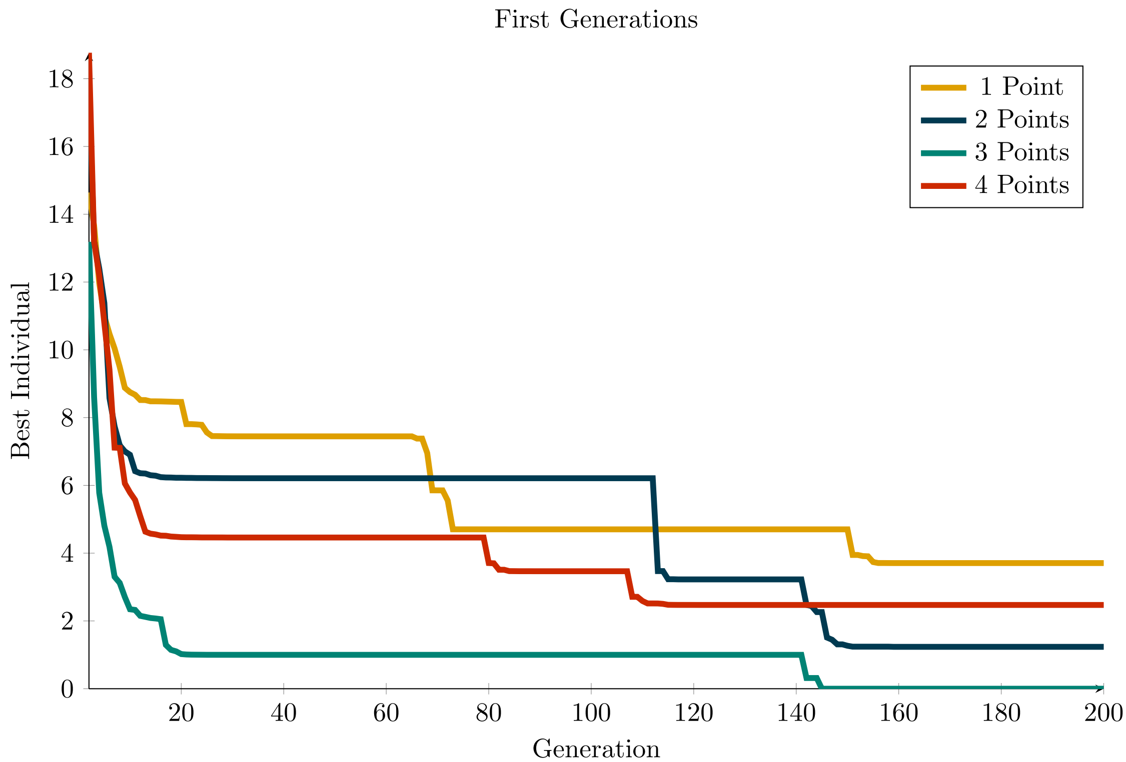


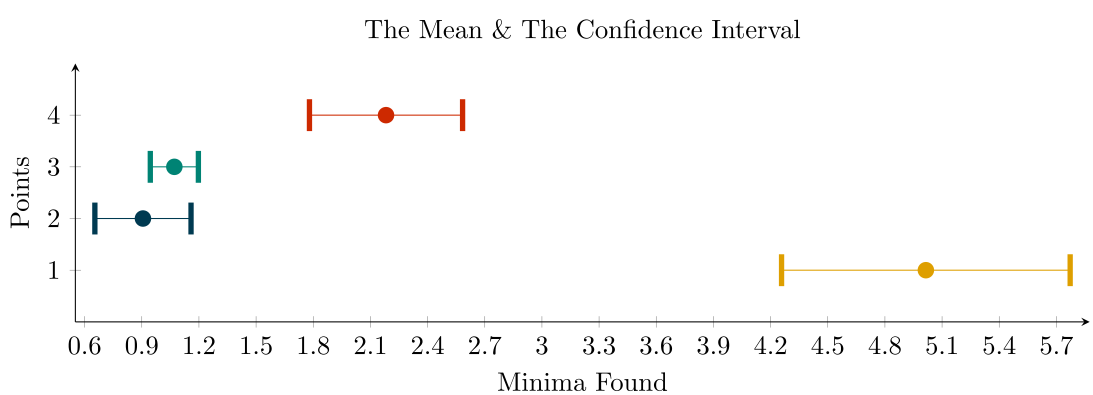
	Mean	MOE	Best	Worst
1 Point	-27.43657	0.07845	-28.19104	-26.51151
2 Points	-26.90755	0.06007	-27.81932	-26.40266
3 Points	-26.24511	0.09141	-27.14284	-25.32027
4 Points	-26.82761	0.09710	-27.81050	-25.59996

Figure 22: Michalewicz - 30 dimensions - Roulette Wheel Selection

2.4 Rastrigin's Function

2.4.1 5 dimensions - Truncate Selection

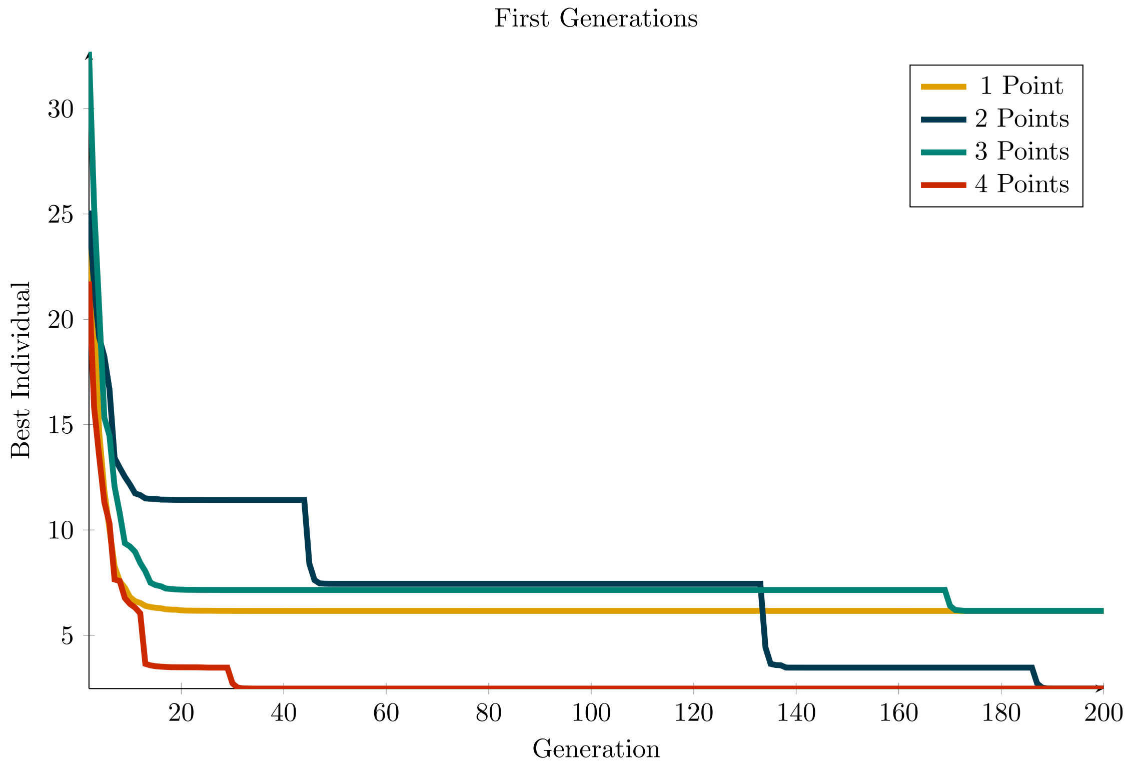


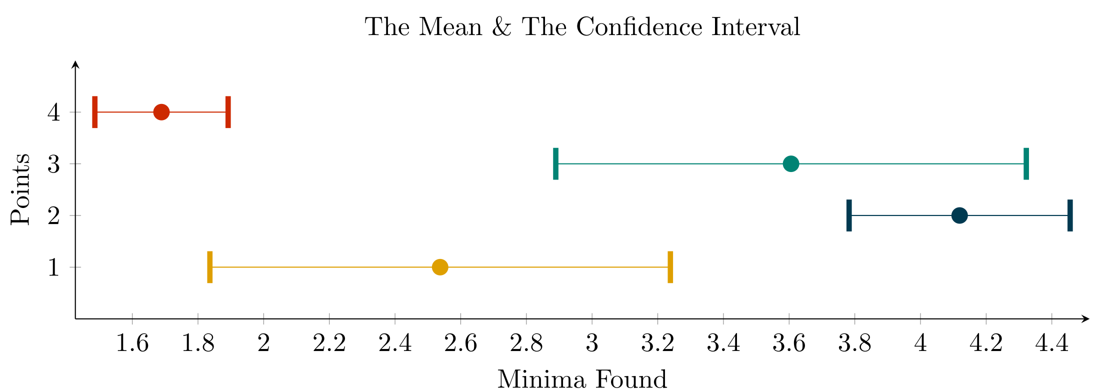


	Mean	MOE	Best	Worst
1 Point	5.01494	0.46038	1.23577	7.39489
2 Points	0.90624	0.15339	0.00000	2.47155
3 Points	1.07100	0.07670	0.00001	1.23577
4 Points	2.18188	0.24426	0.00001	6.15911

Figure 23: Rastrigin - 5 dimensions - Truncation Selection

2.4.2 5 dimensions - Roulette Wheel Selection

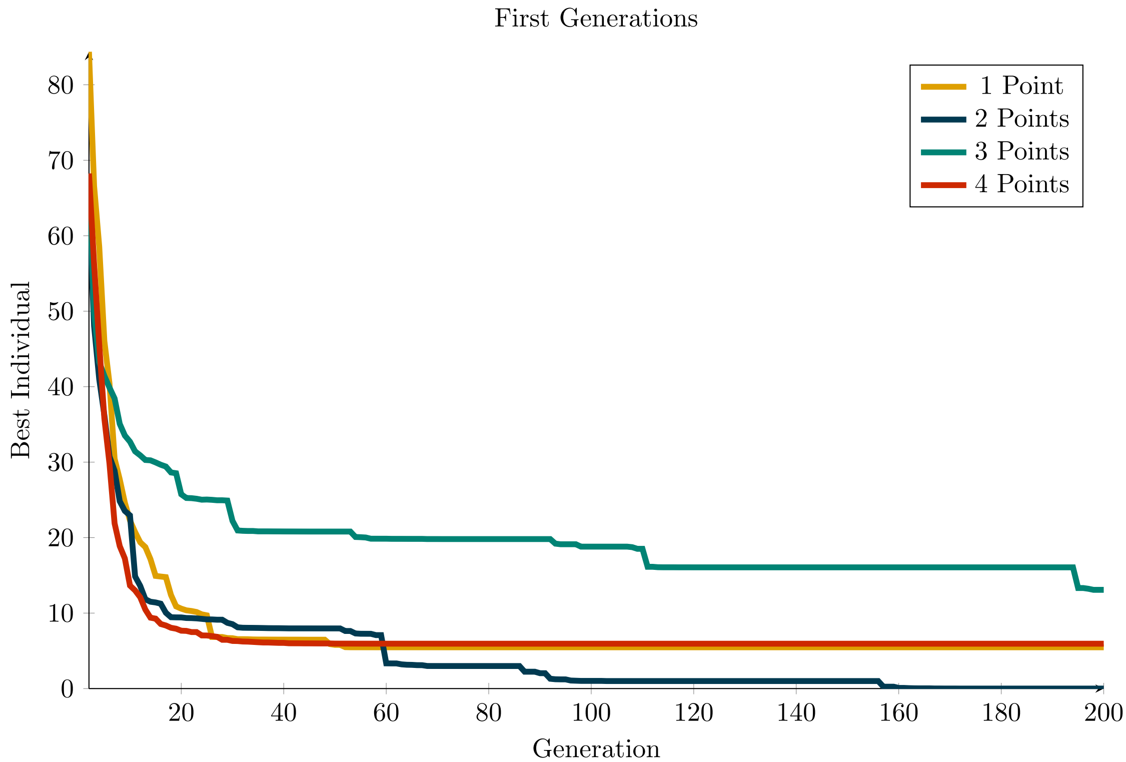


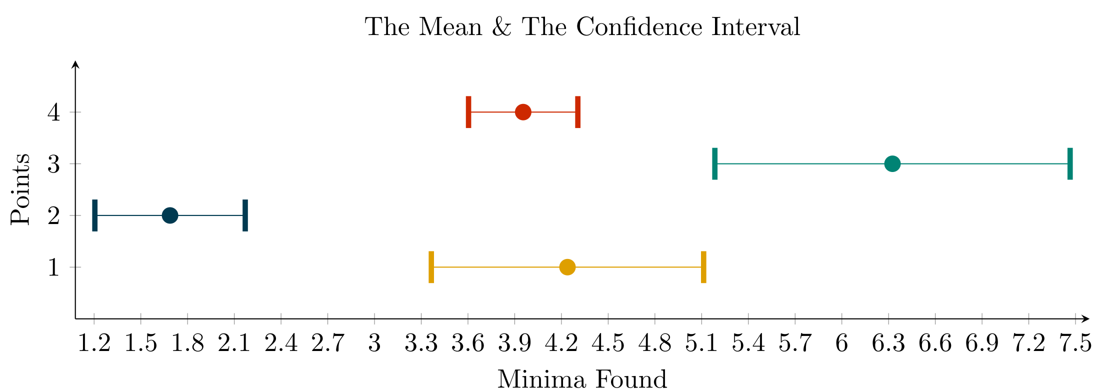


	Mean	MOE	Best	Worst
1 Point	2.53735	0.42622	0.00000	6.15912
2 Points	4.11924	0.20458	2.47154	6.17885
3 Points	3.60588	0.43549	0.00000	6.15912
4 Points	1.68889	0.12335	1.23577	3.70731

Figure 24: Rastrigin's Function - 5 dimensions - Roulette Wheel Selection

2.4.3 10 dimensions - Truncate Selection

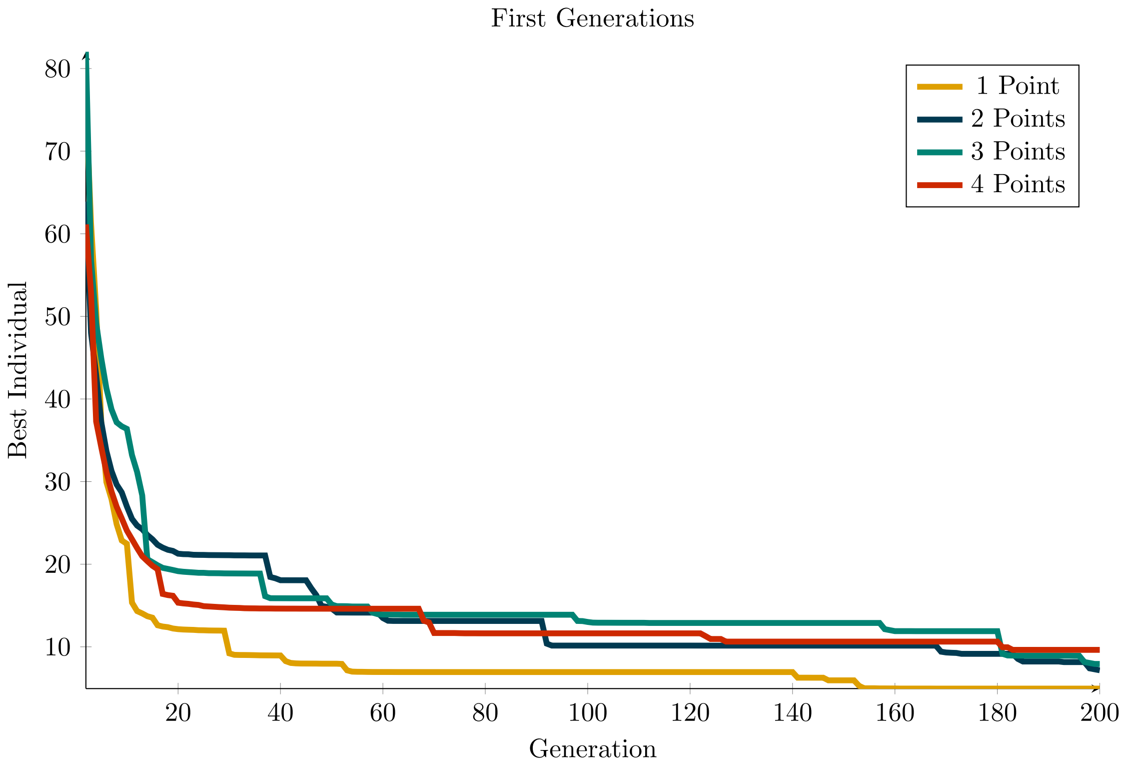


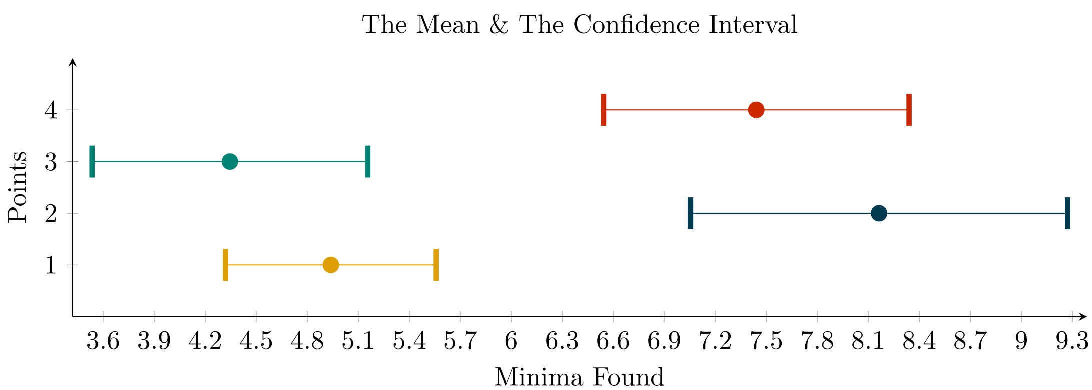


	Mean	MOE	Best	Worst
1 Point	4.23886	0.53136	1.23577	13.55397
2 Points	1.68757	0.29341	0.00000	7.39487
3 Points	6.32519	0.69320	0.00001	12.31821
4 Points	3.95447	0.21325	1.23578	6.17885

Figure 25: Rastrigin - 10 dimensions - Truncation Selection

2.4.4 10 dimensions - Roulette Wheel Selection

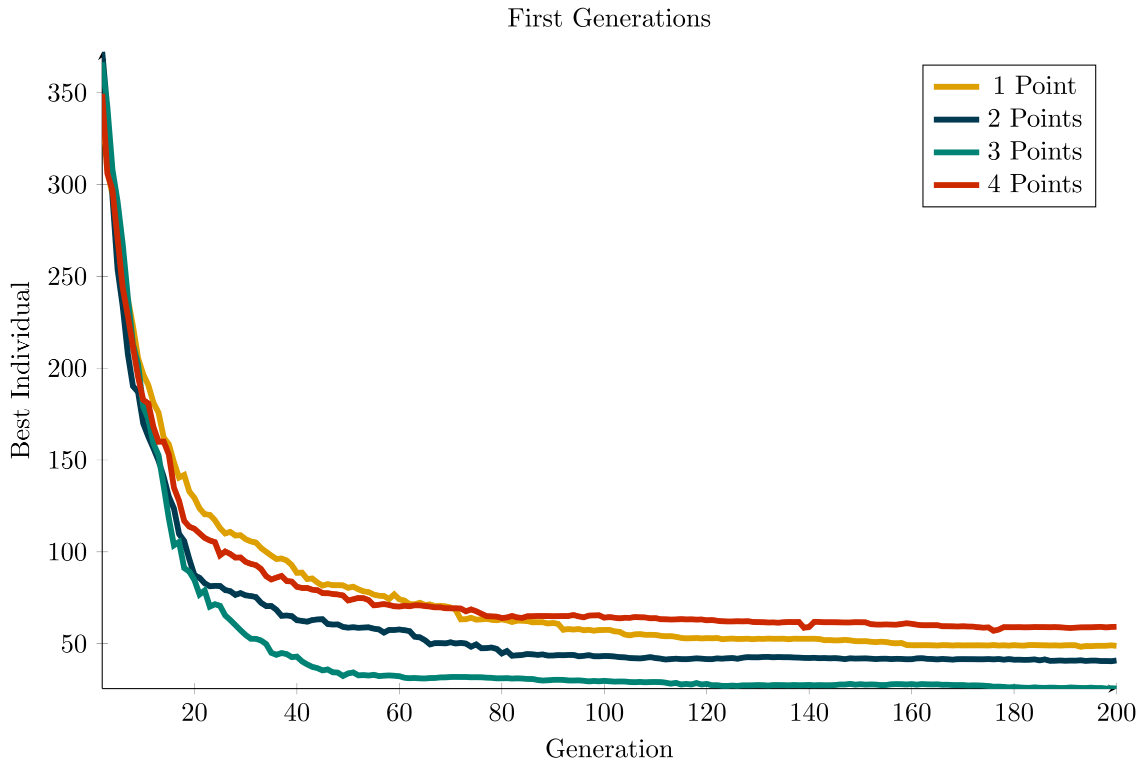


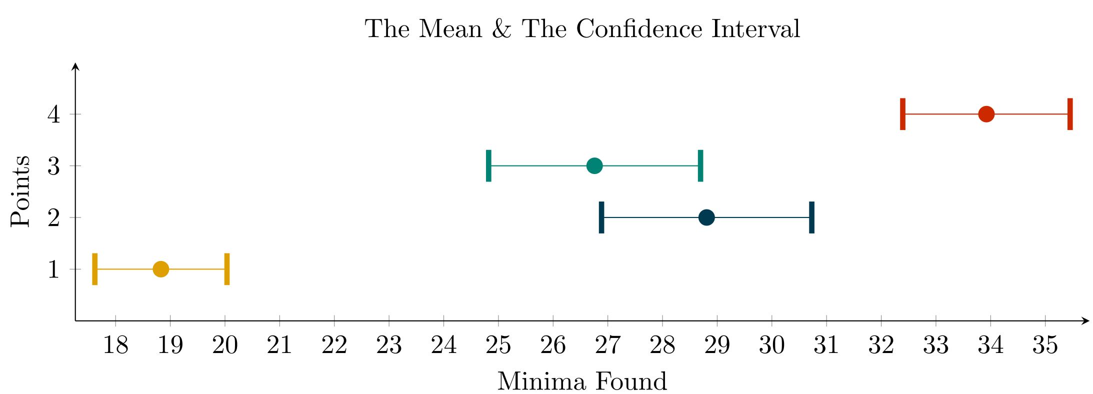


	Mean	MOE	Best	Worst
1 Point	4.93914	0.37631	2.47155	9.86643
2 Points	8.16359	0.67354	0.00001	13.55399
3 Points	4.34537	0.49262	0.00001	11.08245
4 Points	7.44202	0.54593	2.47156	14.78978

Figure 26: Rastrigin's Function - 10 dimensions - Roulette Wheel Selection

2.4.5 30 dimensions - Truncate Selection

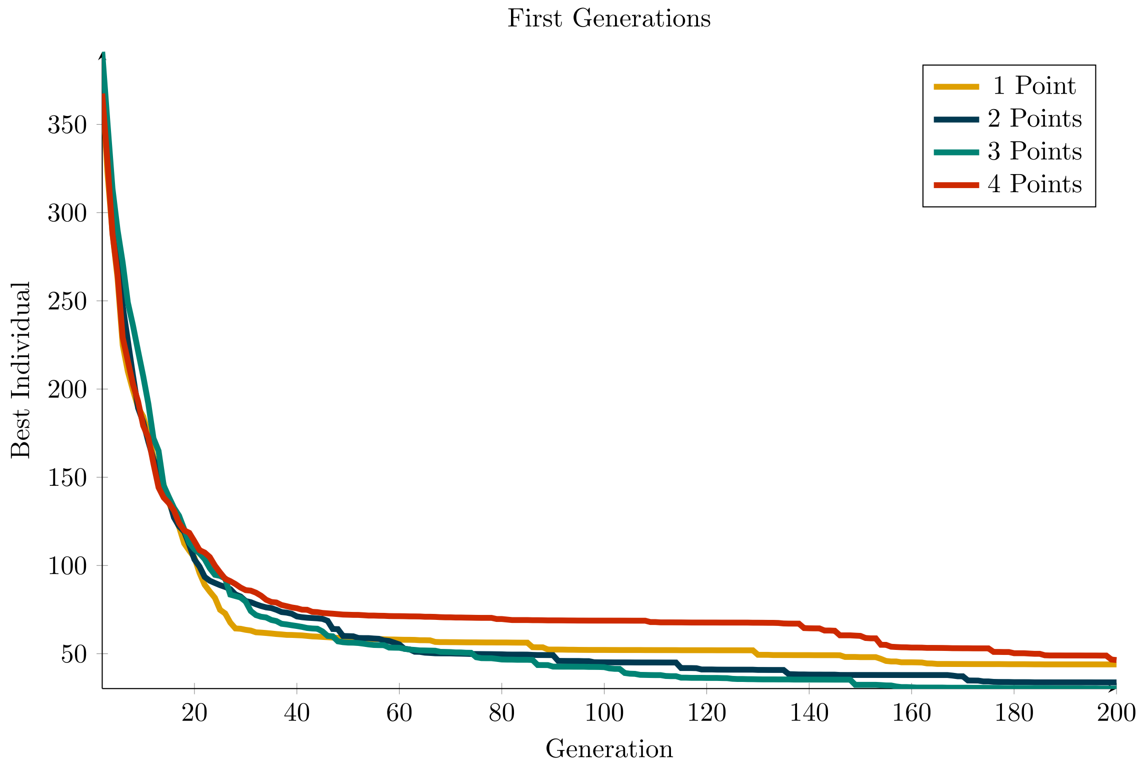


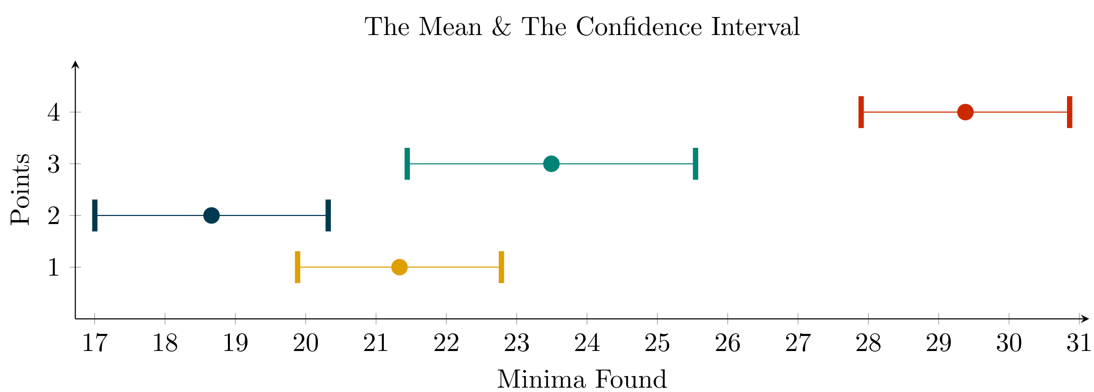


	Mean	MOE	Best	Worst
1 Point	18.82891	0.73442	11.59440	28.54611
2 Points	28.80749	1.16849	15.94313	44.33215
3 Points	26.75862	1.17765	15.44905	40.53295
4 Points	33.92347	0.93031	24.11274	44.96314

Figure 27: Rastrigin's Function - 30 dimensions - Truncation Selection

2.4.6 30 dimensions - Roulette Wheel Selection





	Mean	MOE	Best	Worst
1 Point	21.33423	0.88072	13.11910	33.79377
2 Points	18.66132	1.00862	9.41183	35.34852
3 Points	23.49372	1.24591	13.87607	38.98442
4 Points	29.37984	0.90101	17.58266	38.99331

Figure 28: Rastrigin's Function - 30 dimensions - Roulette Wheel Selection

3 Heuristic Algorithms and Genetic Algorithms Comparison

3.1 De Jong 1 Function

1. *RWS works better compared to Truncation Selection(TS)*

   True : Every algorithm finds a global minimum close to 0 but, for 30 dimensions, is clear that RWS is more consistent.

2. *Higher dimensions need more points in the k-point crossover*

   Cannot Say : The function is simple and the results are similar.

3. *The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner*

   Cannot Say : The function is simple and the results are similar.

3.2 Schwefel's Function

1. *RWS works better compared to Truncation Selection(TS)*

   True : In 5 dimensions, both strategies give similar results, but in 10 and 30, RWS has a lower average.

2. *Higher dimensions need more points in the k-point crossover*

   False : For this function, 3-point or 4-point crossover gives the best result for all dimensions.

3. *The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner*

   True : For 30 dimensions, The best GA (Roulette Wheel Selection with 3-Point Crossover with -12330.16227) has a much lower average value when compared to the best HA (Hill Climbing Best Improvement with -11378.970798).

3.3 Michalewicz's Function

1. *RWS works better compared to Truncation Selection(TS)*

   True : The difference is clear for 30 dimensions, where the best RWS (-27.43657) has a lower minimum found compared to the best TS (-27.36678).

2. *Higher dimensions need more points in the k-point crossover*

   False : This function is the perfect counterexample, at least for dimensions lower or equal to 30. The tests show that 1-Point Crossover and 2-Points Crossover dominate in finding the best global optima.

3. *The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner*

   True : For 30 dimensions, The best GA (Roulette Wheel Selection with 1-Point Crossover with -27.43657) has a much lower average value when compared to the best HA (Hill Climbing Best Improvement with -27.07615).

3.4 Rastrigin's Function

1. *RWS works better compared to Truncation Selection(TS)*

   True : TS finds a lower average value for 5 and 10 dimensions, but 30, a much more complex function, RWS finds a lower average minimum.

2. *Higher dimensions need more points in the k-point crossover*

   False : The number of points in the k-point crossover is not directly proportional to the number of dimensions.

3. *The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner*

   True : For 30 dimensions, The best GA (Roulette Wheel Selection with 2-Point Crossover with 18.66132) has a much lower average value compared to the best HA (Hill Climbing Best Improvement with 28.633780).

4 Conclusion

1. *RWS works better compared to Truncation Selection(TS), at least for wave functions*

   True : RWS finds a lower average value for most functions in 5 and 10 dimensions, but for 30 dimensions, RWS has a lower optimum for every benchmark function.

2. *Higher dimensions need more points in the k-point crossover*

   False : The number of points in the k-point crossover is not directly proportional to the number of dimensions.

3. *The most noticeable difference between GAs and Heuristic Algorithms (HA) is between functions of higher dimension, where the GA is the winner*

   True : GAs find a lower optimum compared to any HA in any dimension.

4. *HAs work better for paraboloid types of functions due to their Hill Climbing Nature*

   Cannot Say : The function is simple and the results are similar.

References

[Bal96] William H Baltosser. Biostatistical analysis. *Ecology*, 77(7):2266–2268, 1996.

[BFM97] Thomas Bäck, David B Fogel, and Zbigniew Michalewicz. Handbook of evolutionary computation. *Release*, 97(1):B1, 1997.

[BFM18] Thomas Bäck, David B Fogel, and Zbigniew Michalewicz. *Evolutionary computation 1: Basic algorithms and operators*. CRC press, 2018.

[Bu1] George Buțco. Differences between hill climbing algorithm (first improvement, best improvement) and simulated annealing algorithm in finding global minimum of numeric functions. https://github.com/George-debug/Uni-projects/blob/main/GA/T1/T1.pdf, 2021.

[DJ75] Kenneth De Jong. An analysis of the behavior of a class of genetic adaptive systems. 01 1975.

[JY13] Momin Jamil and Xin-She Yang. A literature survey of benchmark functions for global optimisation problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, 4(2):150–194, 2013.

[McC] James D. McCaffrey. Plotting schwefel's function with scilab. https://jamesmccaffrey.wordpress.com/2011/12/10/plotting-schwefels-function-with-scilab/.

[Mit98] Melanie Mitchell. *An introduction to genetic algorithms*. MIT press, 1998.

[Moo] David Moore. A 2d test function known as the michalewicz function (michalewicz 1998). https://www.researchgate.net/figure/12-A-2D-test-function-known-as-the-Michalewicz-function-Michalewicz-1998_fig6_277197330.

[Ras74] L. A. Rastrigin. *Systems of extremal control*. Mir, Moscow, 1974.

[Sam76] Jeffrey R Sampson. Adaptation in natural and artificial systems (john h. holland), 1976.

[Wik] Wikipedia. Rastrigin function. https://en.wikipedia.org/wiki/Rastrigin_function.