VLMOPSO Technical Report

Mathematical Functions: Using Mutation in Both MOPSO and VLMOPSO

1. Rosen-Griewanks-Rastrigin:

Objectives:

f(1)=-RosenbrockObjFun(x);

f(2)=Griewanks(x);

f(3)=-Rastrigin(x);

Search Space: [-600 600]

Lower Length: 1 population Size: 200 Iterations: 200

Higher Length: 20

Results:

Set coverage:

VLMOPSO Vs MOPSO

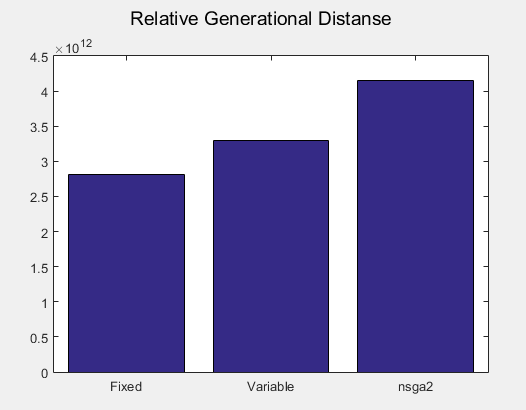
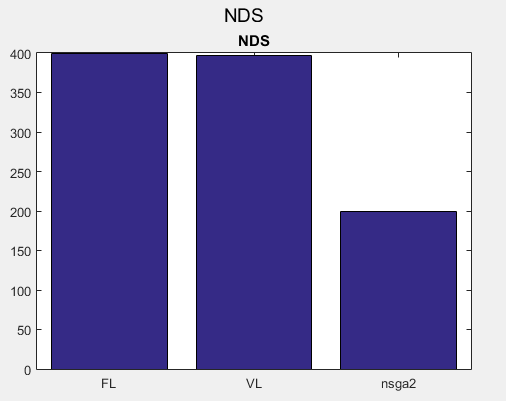
|  |  |
| --- | --- |
| C(vl-mopso,mopso) | C(mopso,vl-mopso) |
| 0.002506 | 0.02 |
| 0.01763 | 0.01763 |
| 0.01508 | 0.007595 |
| 0.01003 | 0.0175 |
| 0.02256 | 0.03008 |
| 0.02256 | 0.002532 |
| 0.0125 | 0.0101 |
| 0.007538 | 0.02015 |
| 0.0075 | 0.005025 |
| 0.005013 | 0.01511 |
| T Test: | 0.529852 |

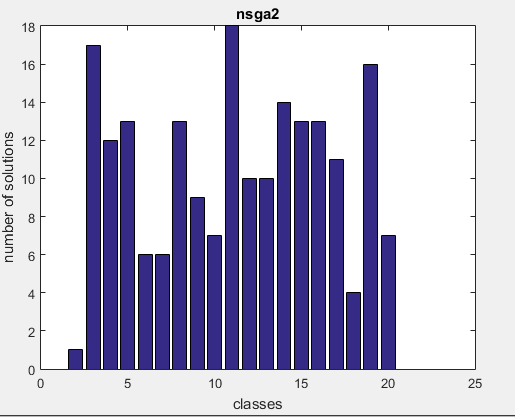
As we see there is no significant difference

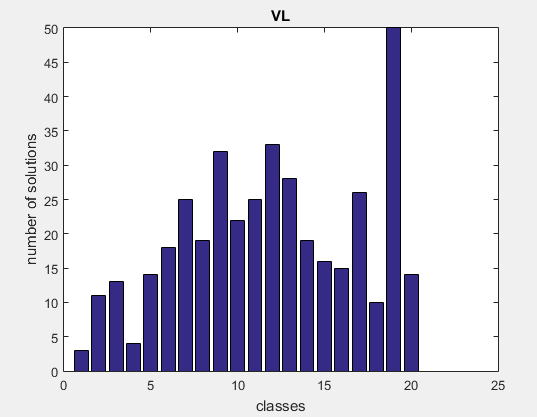
VLMOPSO vs NSGA2

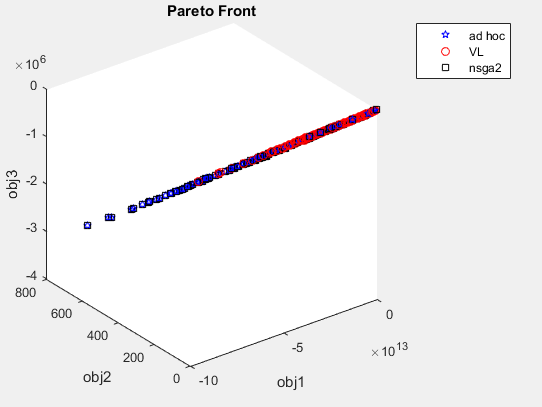
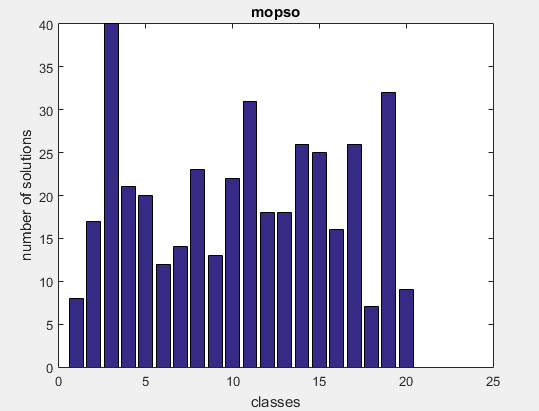
|  |  |
| --- | --- |
| C(vl-mopso,nsga2) | C(nsga2,vl-mopso) |
| 0.015 | 0.0125 |
| 0.03 | 0.01259 |
| 0.015 | 0.005063 |
| 0.025 | 0.0175 |
| 0.025 | 0.02506 |
| 0.025 | 0.002532 |
| 0.025 | 0.005051 |
| 0.02 | 0.01763 |
| 0.02 | 0.005025 |
| 0 | 0.007557 |
| T Test: | 0.017409 |

There is significant difference and vlmopso is better than nsga2









1. Rosen-Rastrigin

Objectives:

f(1)=RosenbrockObjFun(x);

f(2)=-Rastrigin(x);

Search Space: [-5.12 5.12]

Lower Length: 1 population Size: 500 Iterations: 200

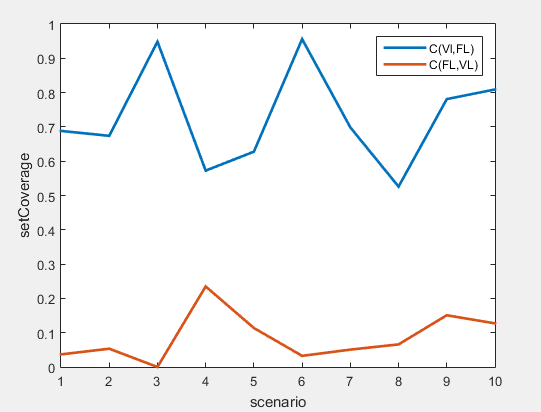
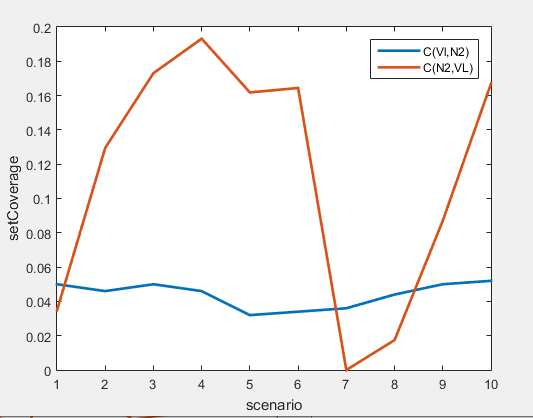
Higher Length: 30

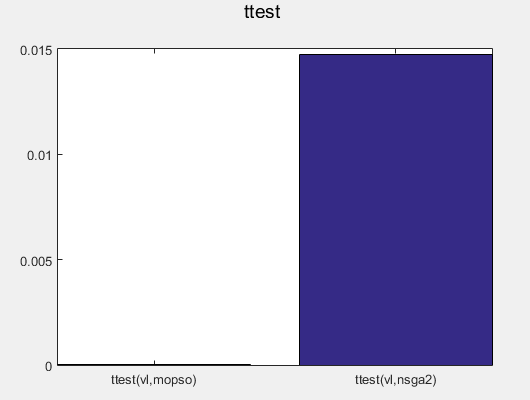
Results:

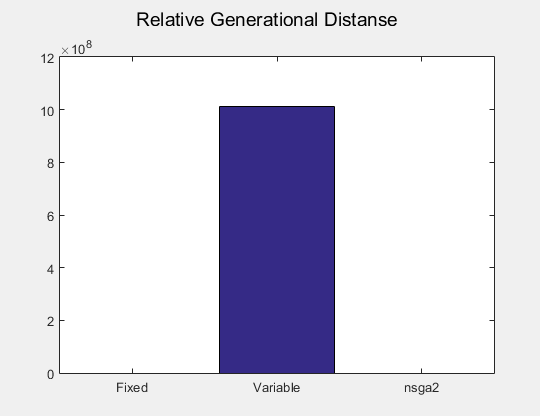
Set coverage:

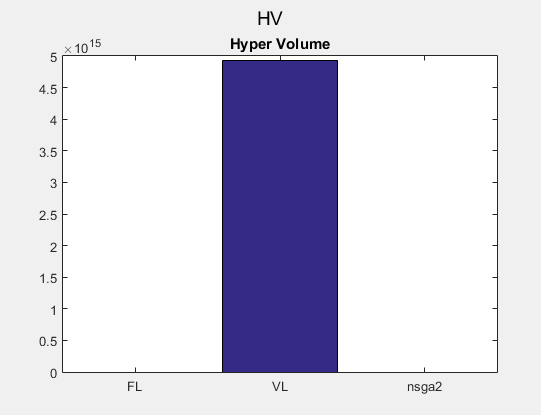
VL = VLMOPSO

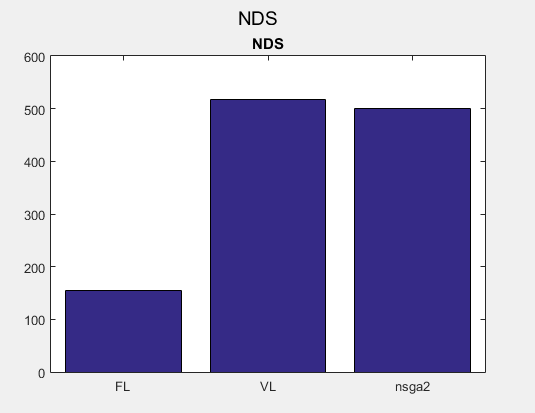
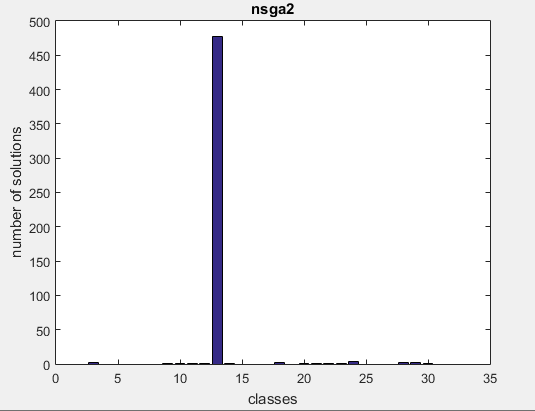
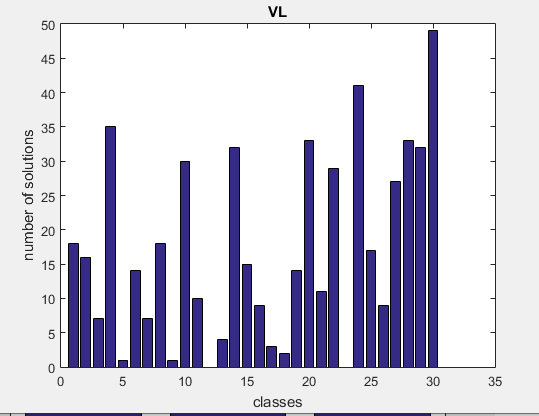
FL= MOPSO

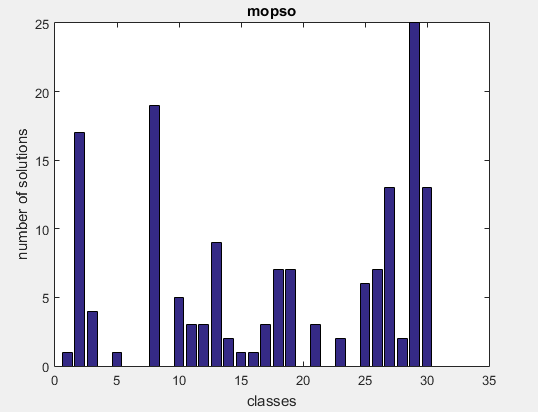
N2=NSGA2

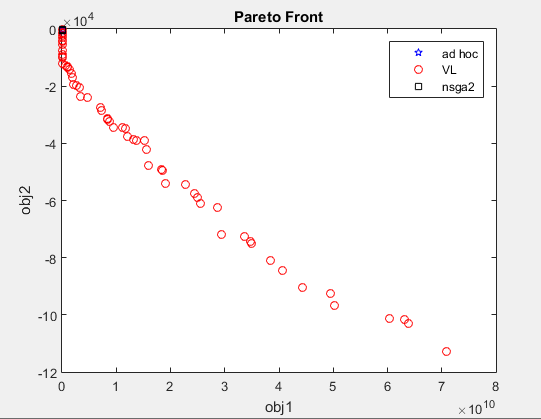
 There is significant Difference











3- levy-stybtang-powell

f(1)=levy(x);

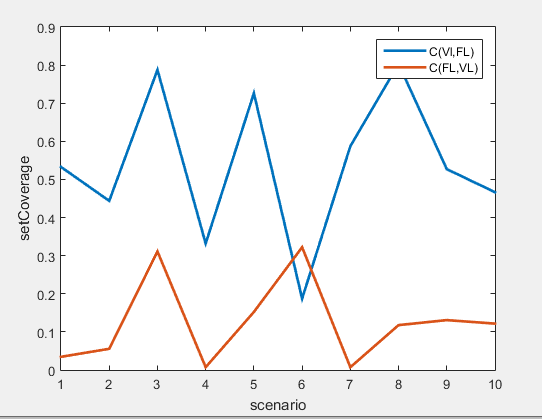
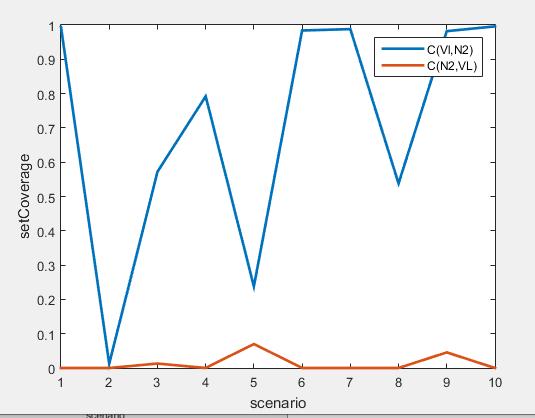
f(2)=powell(x);

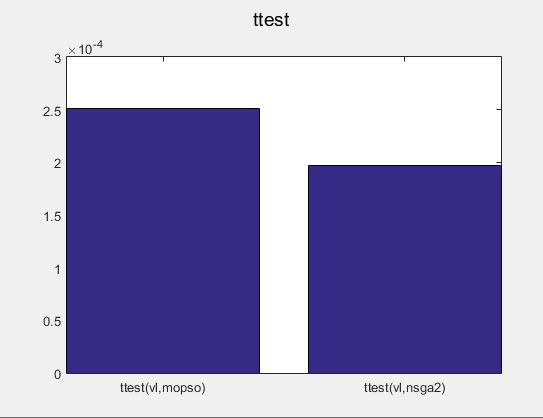
f(3)=stybtang(x); stybtang function is decreasing as length increases

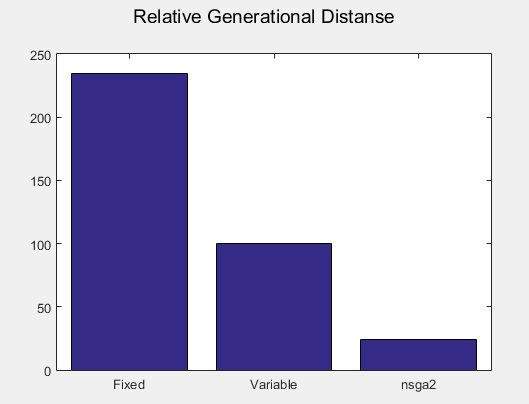
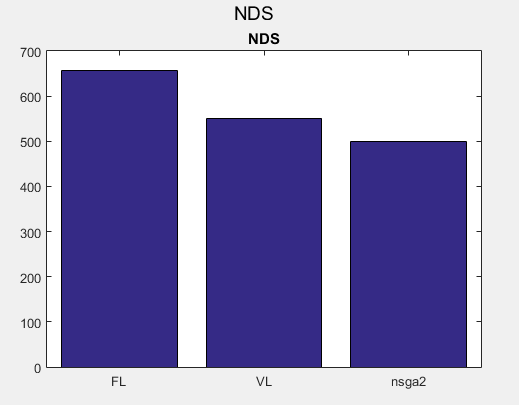
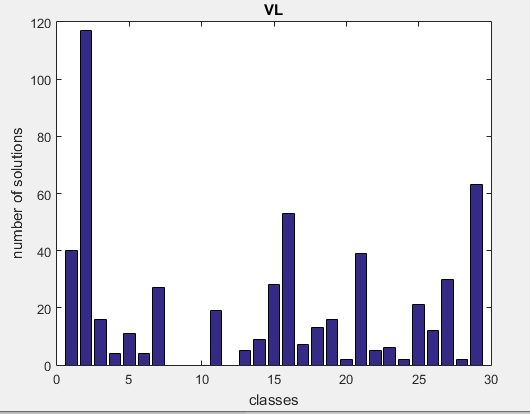
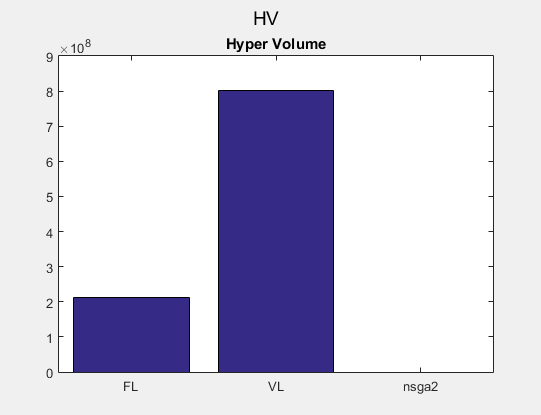
Search Space: [-10 10]

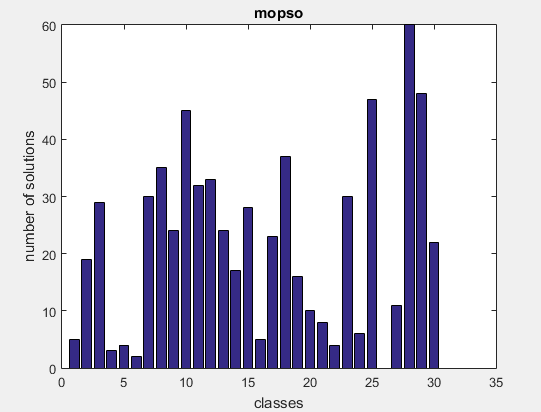
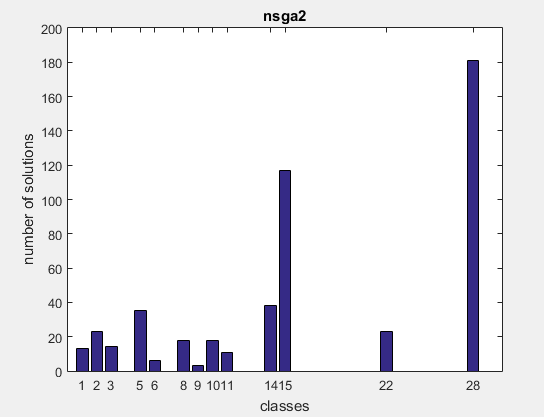
Lower Length: 1 population Size: 500 Iterations: 200

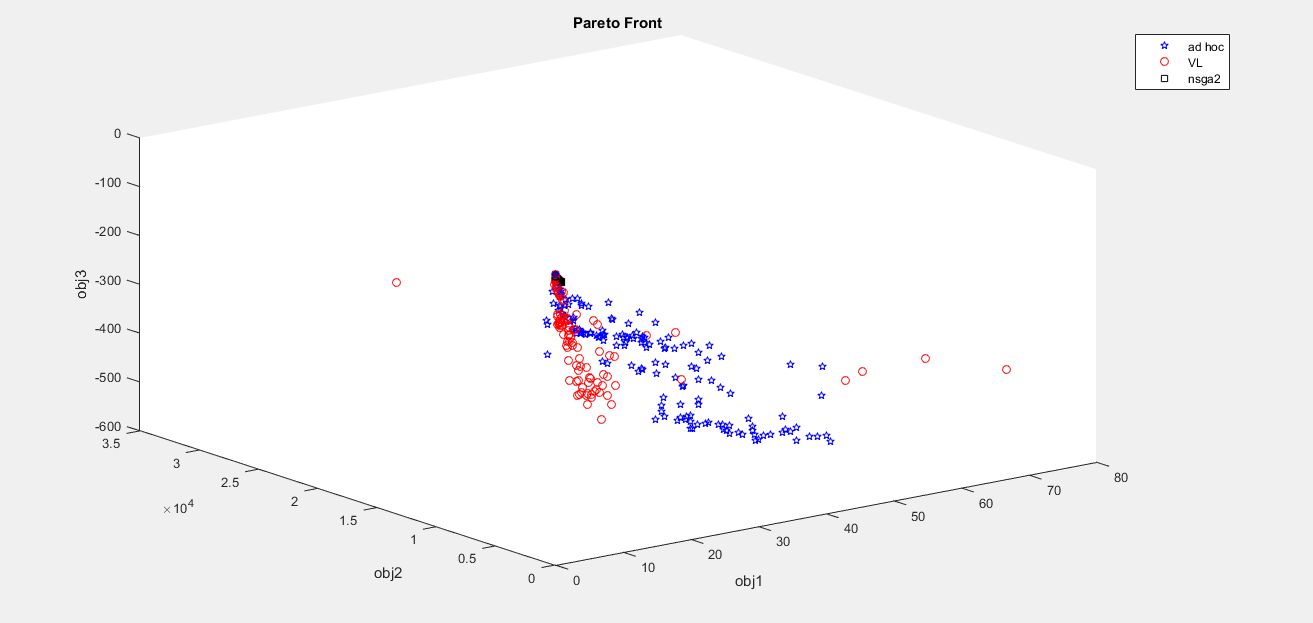
Higher Length: 30

Results:









1. Rosen-Sphere

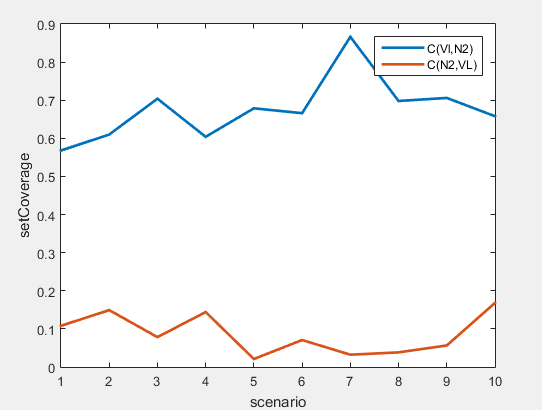
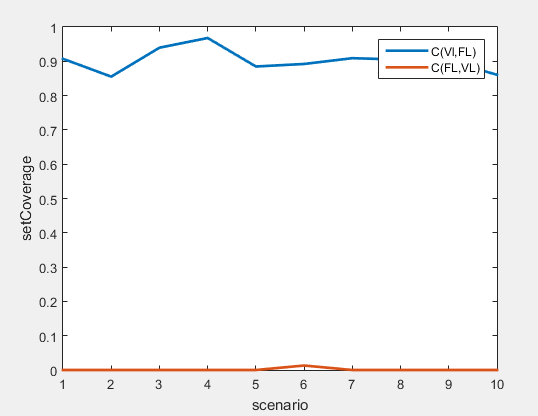
f(1)=RosenbrockObjFun(x);

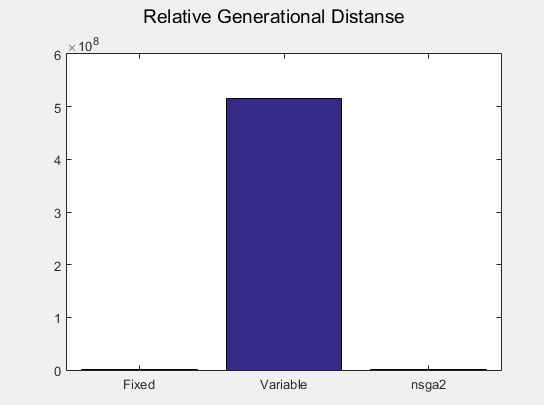
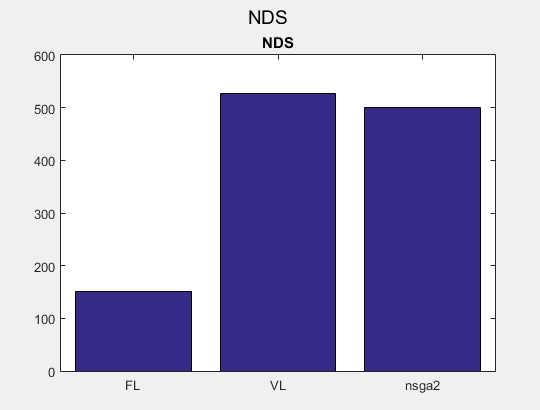
f(2)=-sum(x.^2);

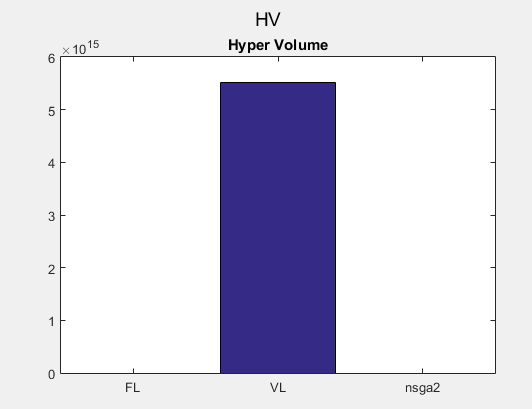
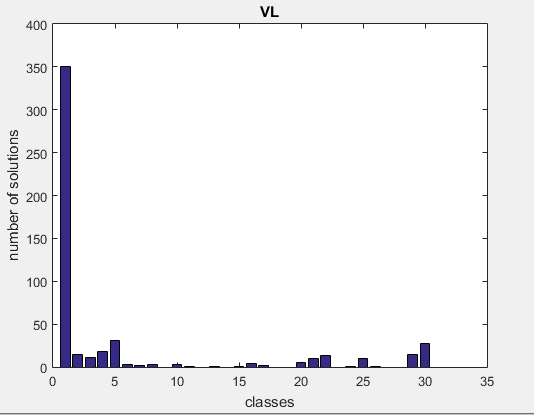
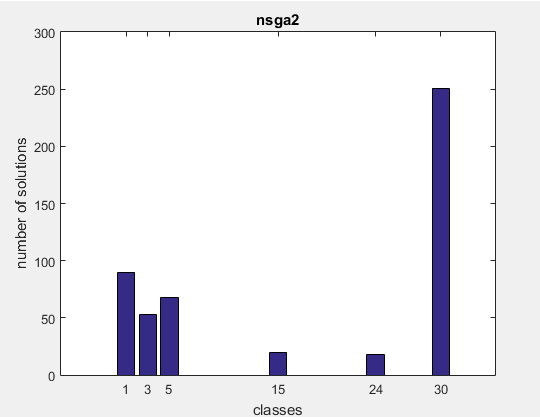
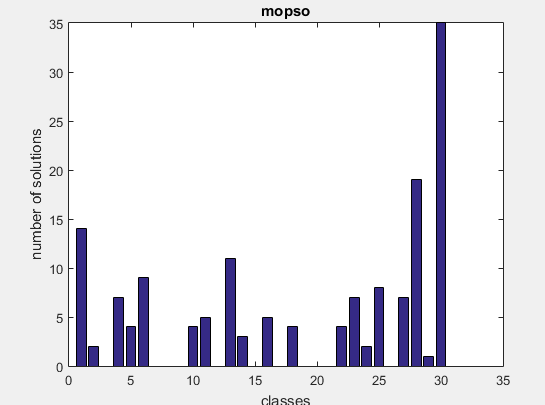
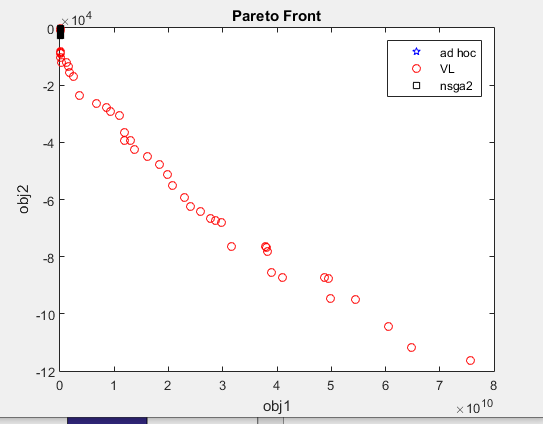
Search Space: [-10 10]

Lower Length: 1 population Size: 500 Iterations: 200

Higher Length: 30

 Results:





5- stybtang\_Griewanks\_Schwefel

Objectives:

f(1)= stybtang(x);

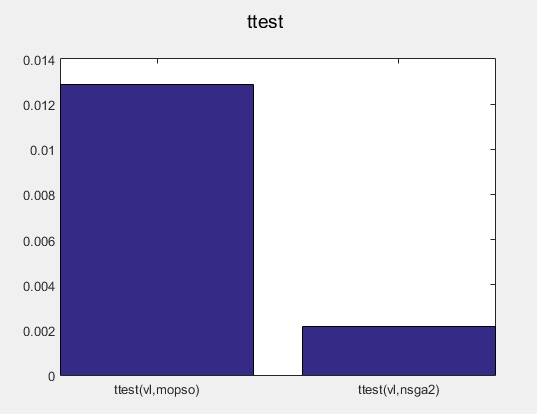
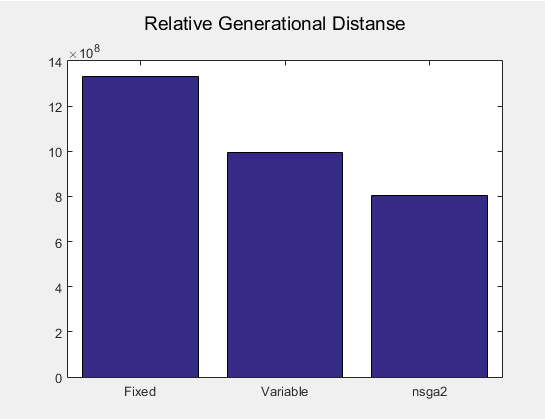
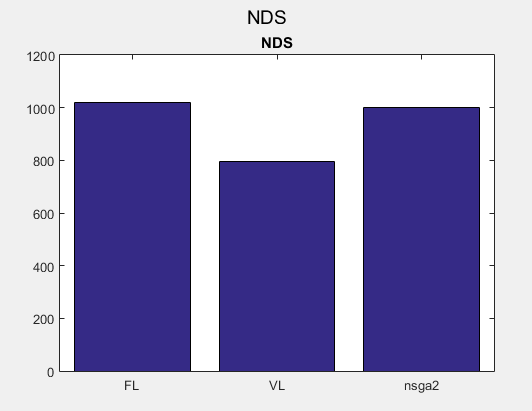
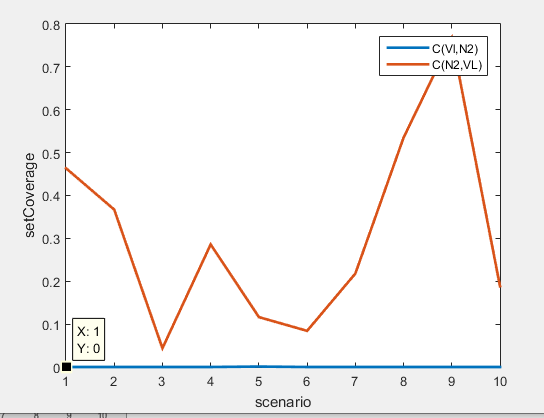
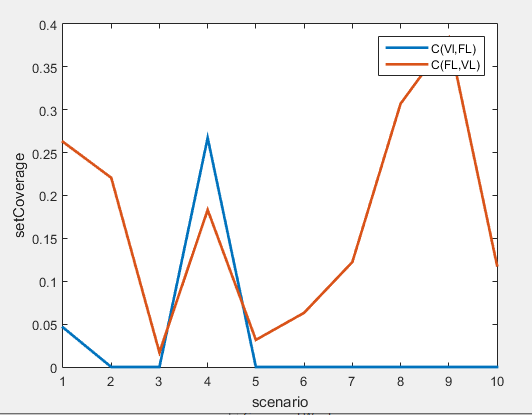
f(2)= Griewanks(x);

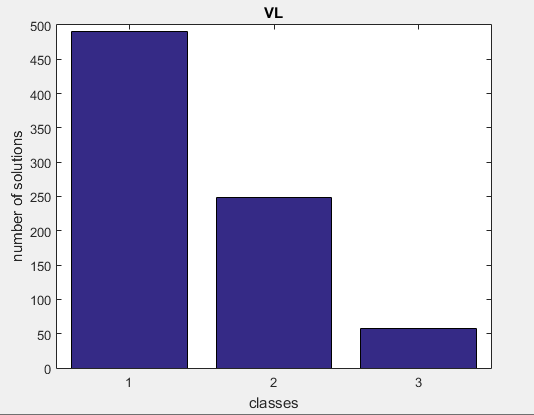
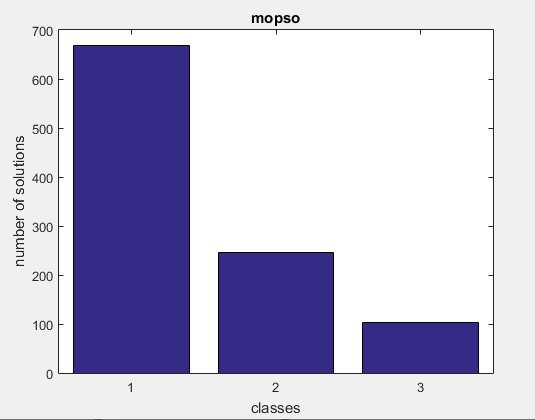
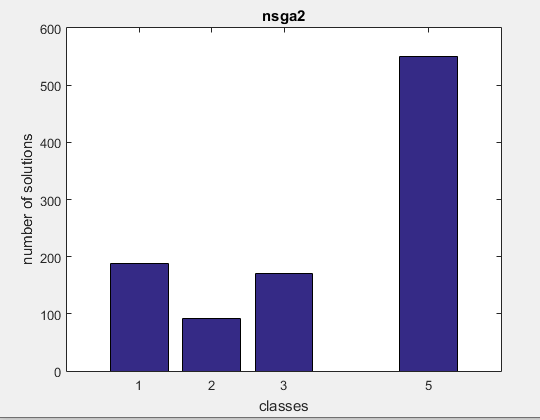
f(3)= Schwefel(x);

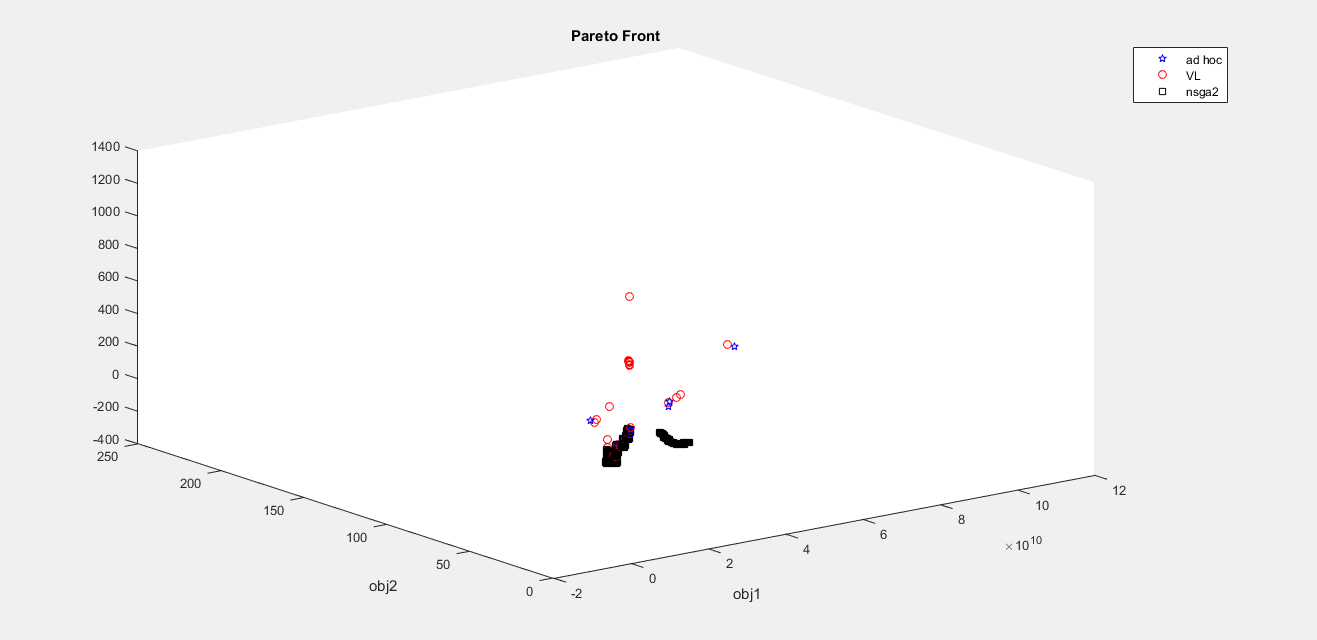
Search Space: [-600 600]

Lower Length: 1 population Size: 1000 Iterations: 200

Higher Length: 30







6- Rastrigin-stybtang

Objectives:

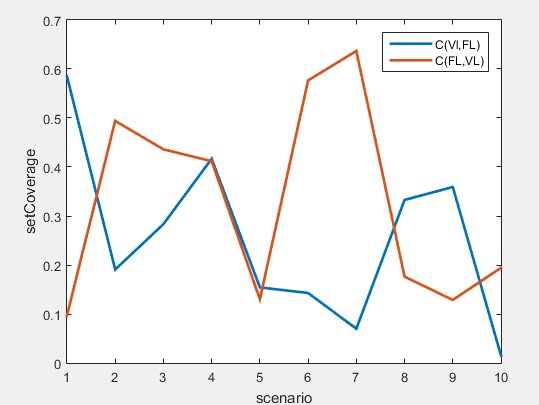
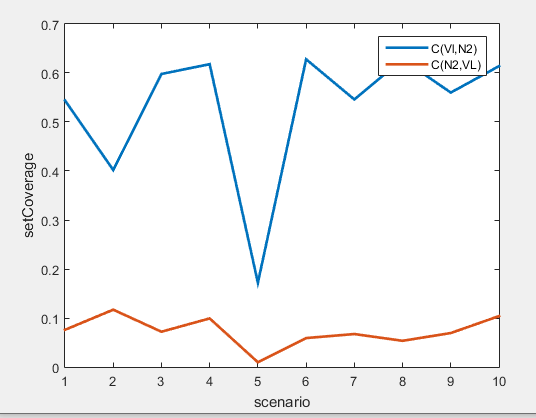
f(1)=Rastrigin(x);

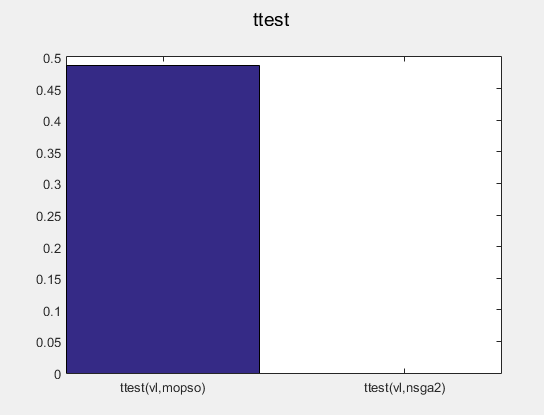
f(2)=stybtang(x);

Search Space: [-5.12 5.12]

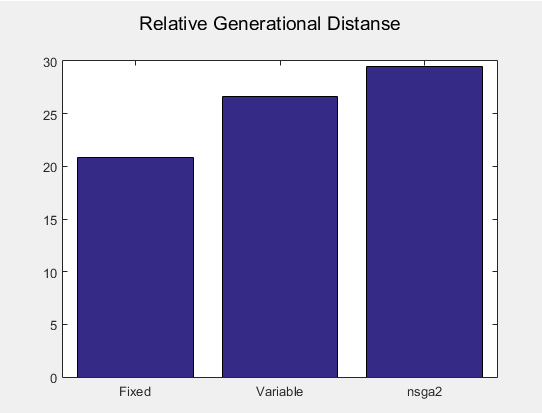
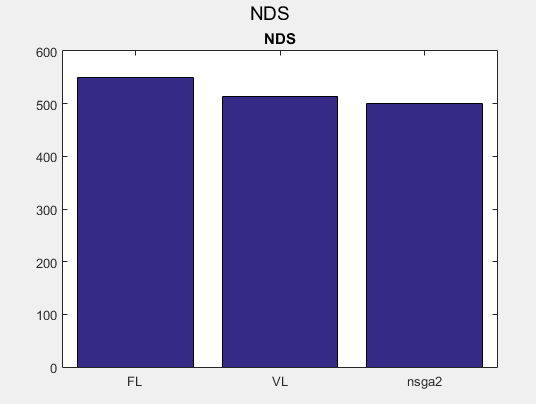
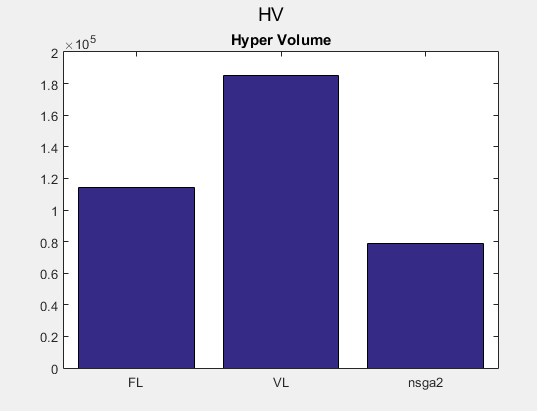
Lower Length: 1 population Size: 500 Iterations: 200

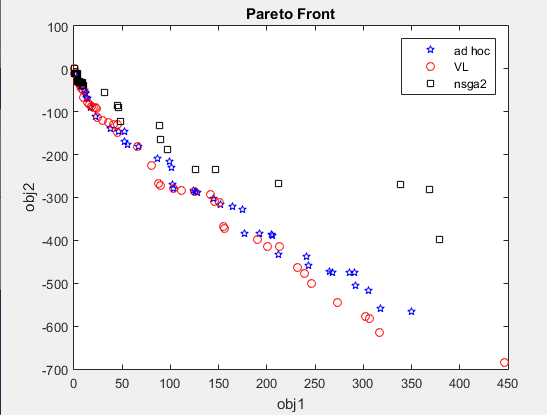
Higher Length: 30

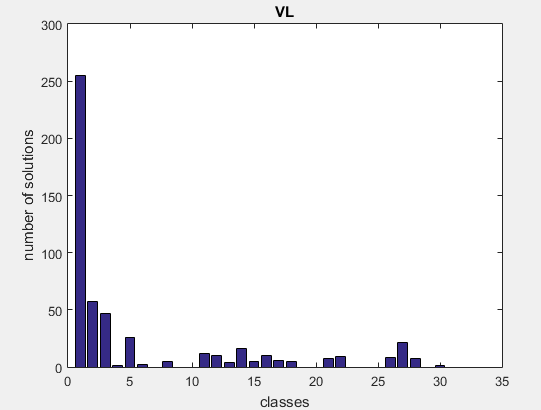
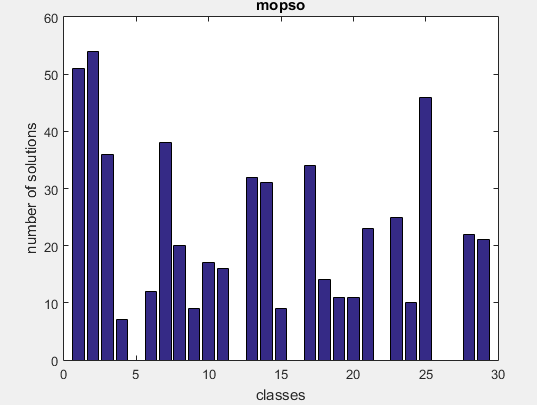


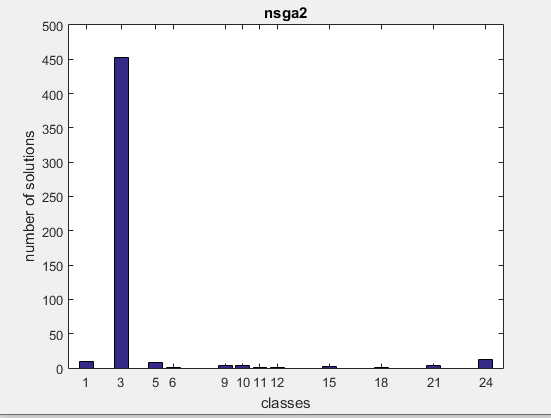


There is no significant difference vetween mopso and vlmopso









7- powell-stybtang-drastrigin

Objectives:

f(1)=powell(x);

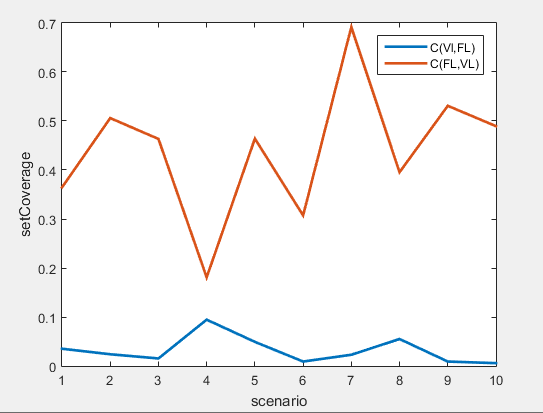
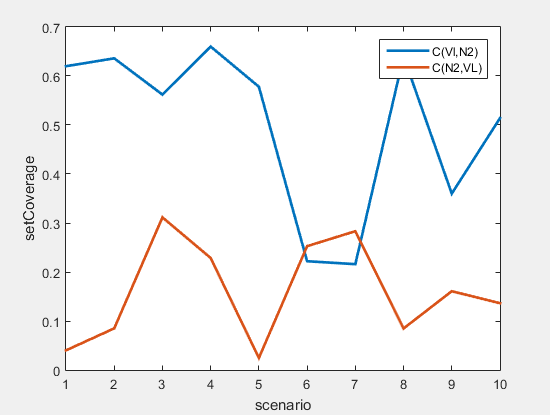
f(2)=stybtang(x);

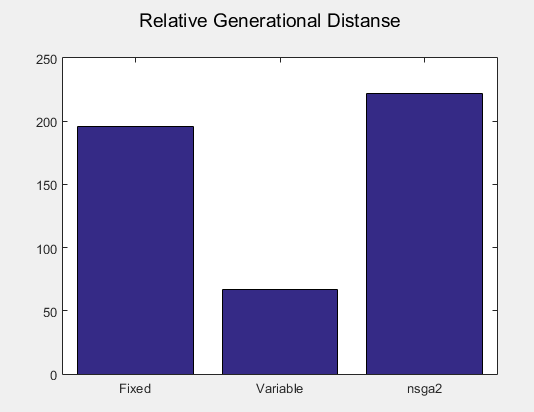
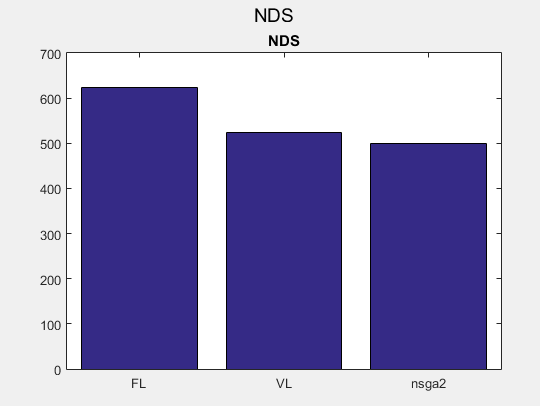
f(3)=DRastrigin(x);

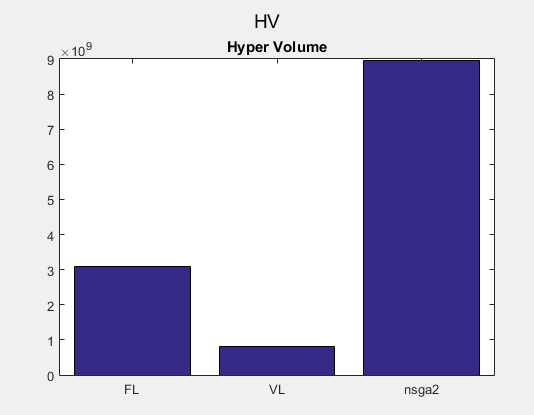
Search Space: [-5.12 5.12]

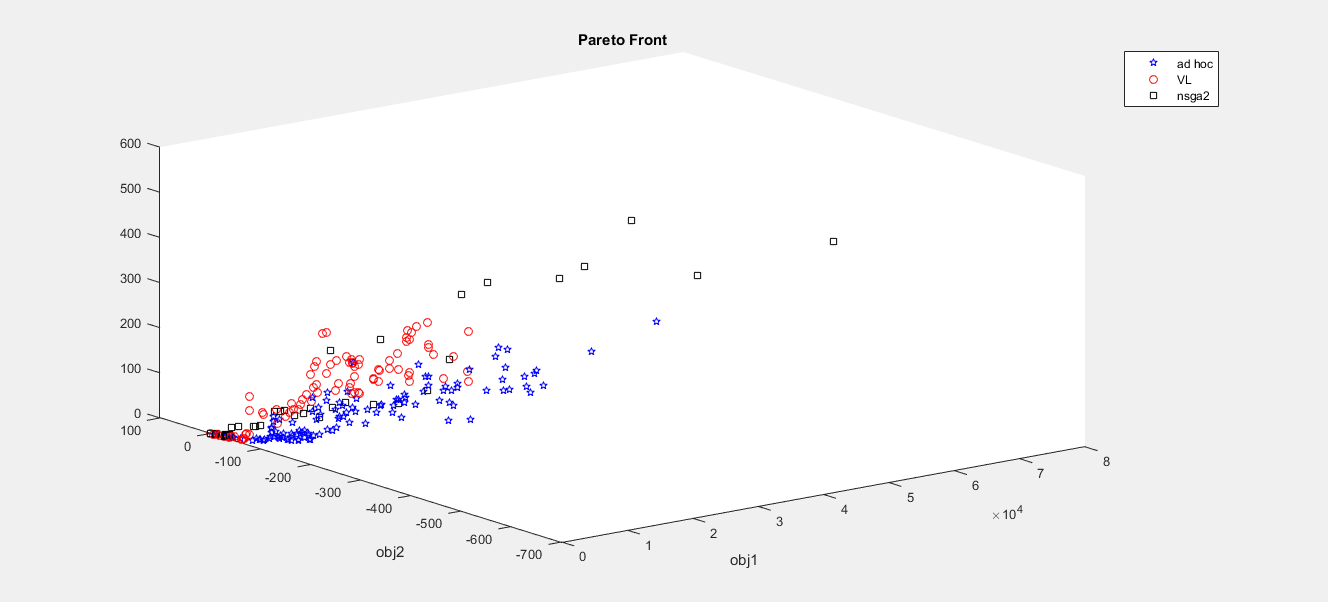
Lower Length: 1 population Size: 500 Iterations: 200

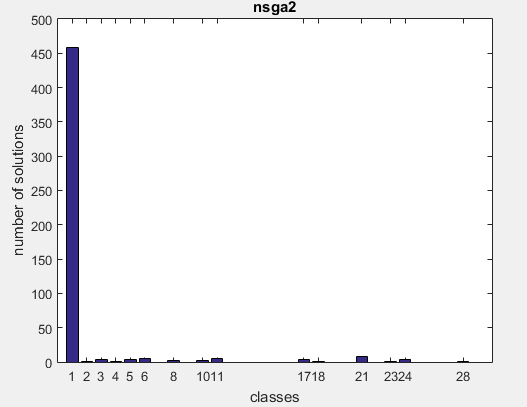
Higher Length: 30

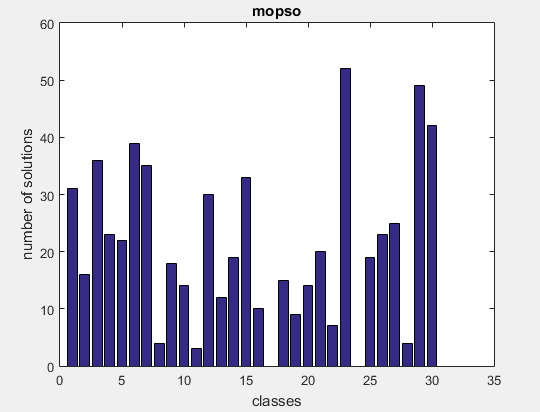


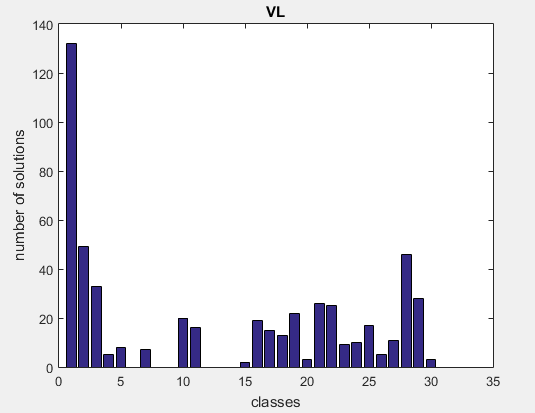












8- Weierstrass-stybtang

Objectives:

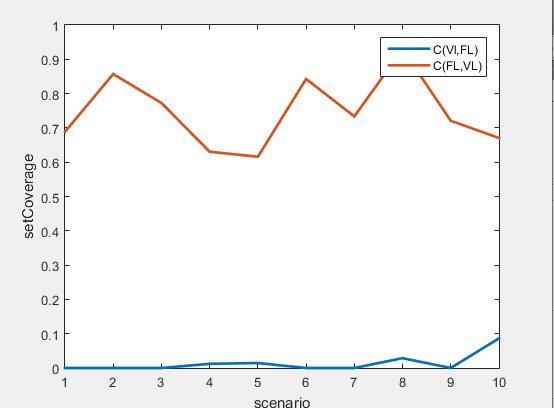
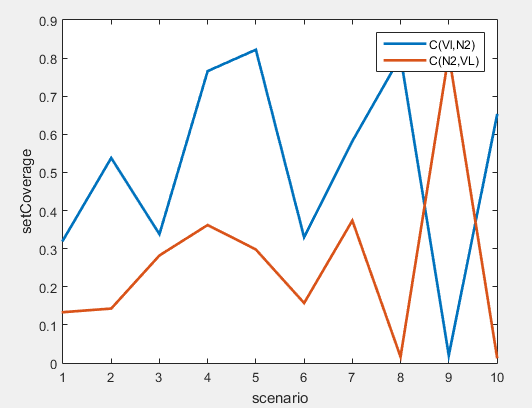
f(1)=Weierstrass(x);

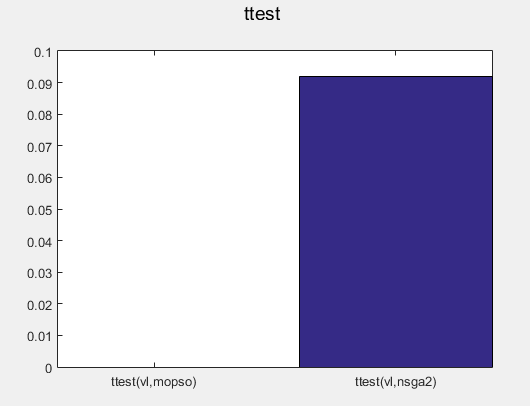
f(2)=stybtang(x);

Search Space: [-5 5]

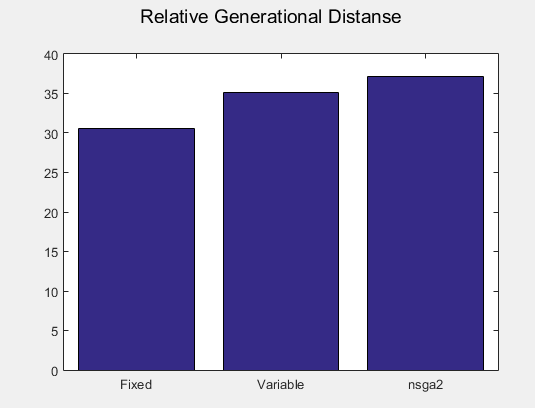
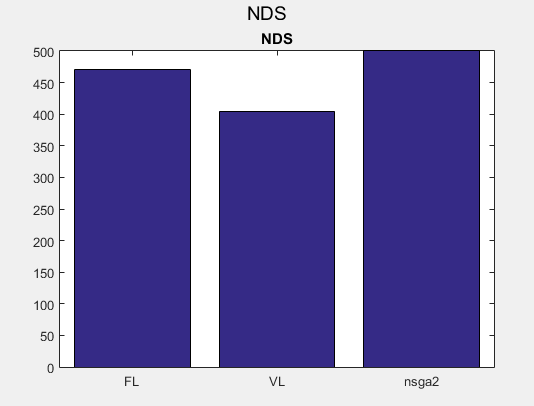
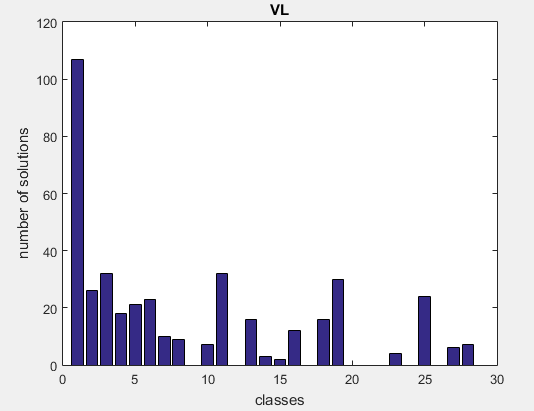
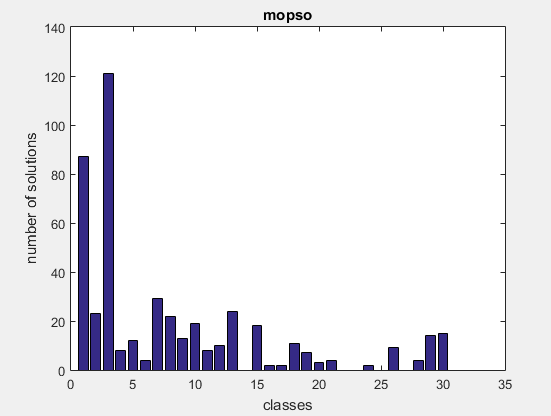
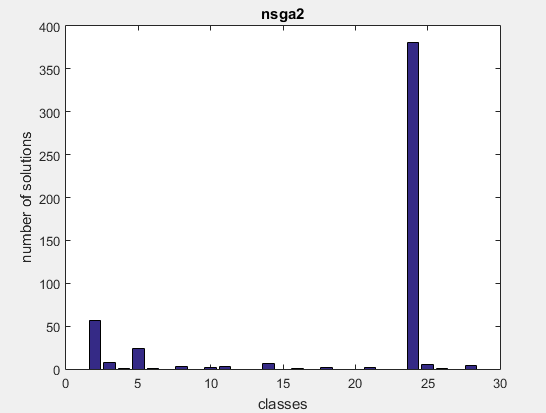
Lower Length: 1 population Size: 500 Iterations: 200

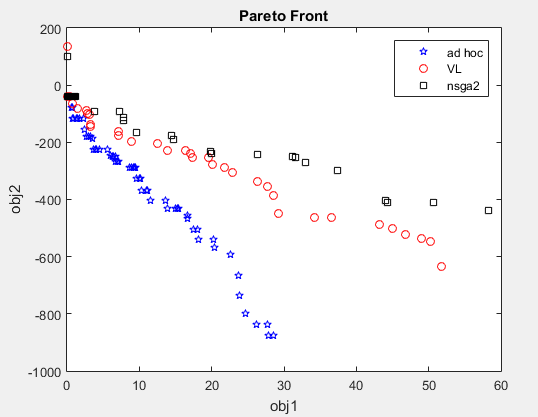
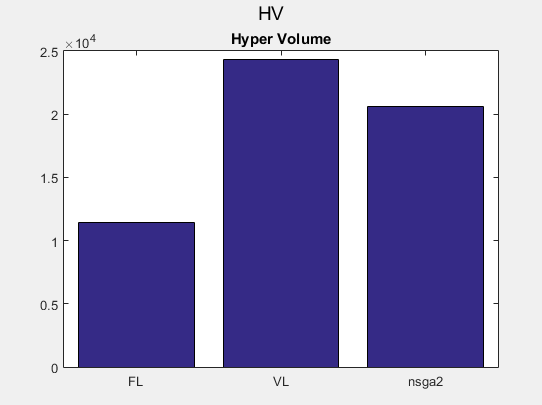
Higher Length: 30





There is no significant difference between vlmopso and nsga2





9- Rosen-Levy

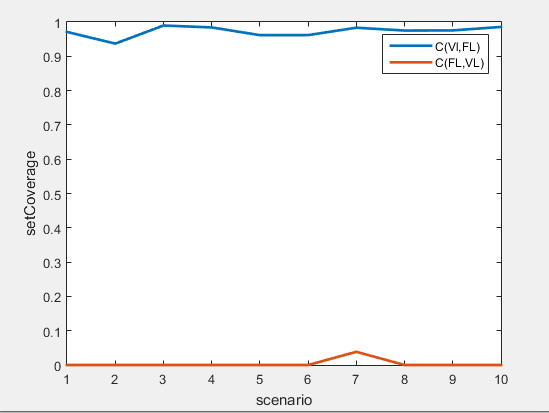
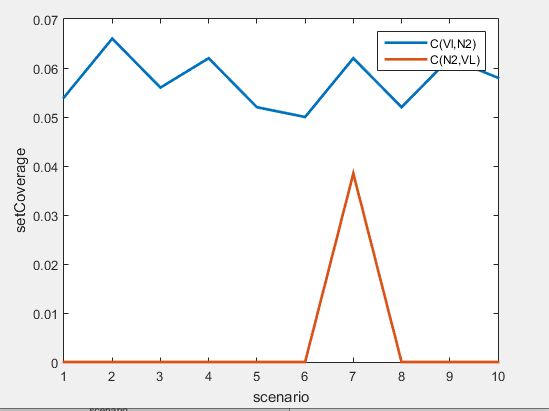
Objectives:

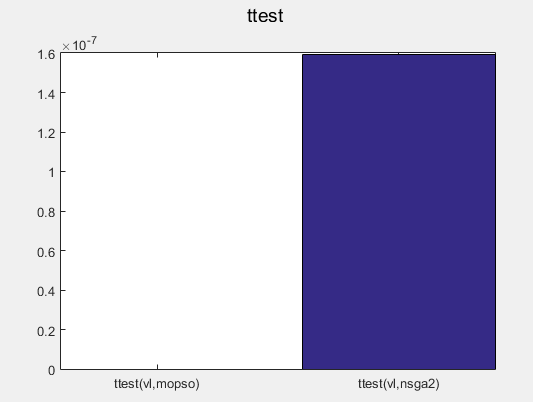
f(1)=RosenbrockObjFun(x);

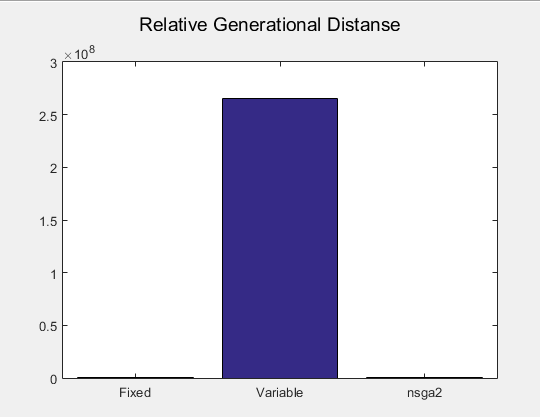
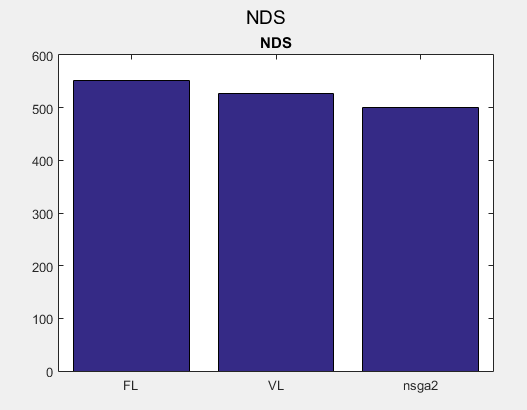
f(2)=-levy(x);

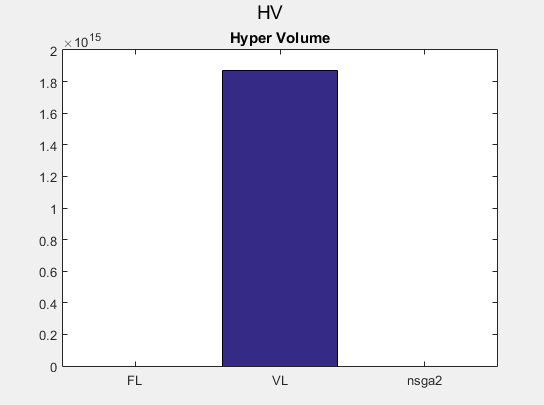
Search Space: [-10 10]

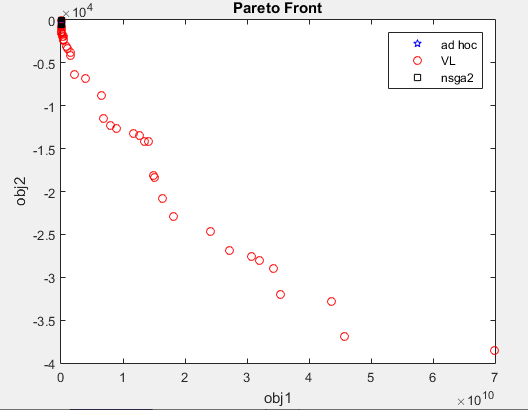
Lower Length: 1 population Size: 500 Iterations: 200

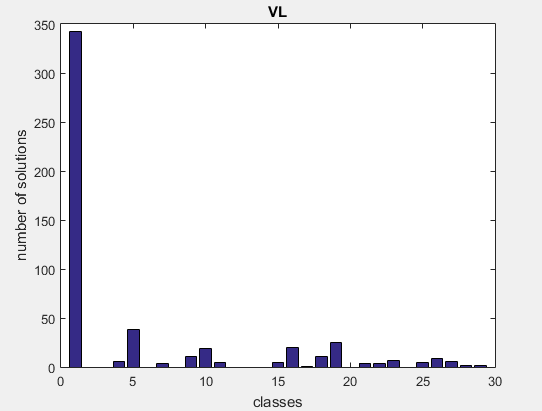
 Higher Length: 30

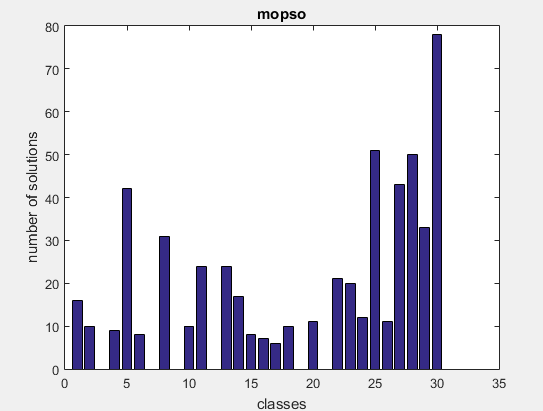


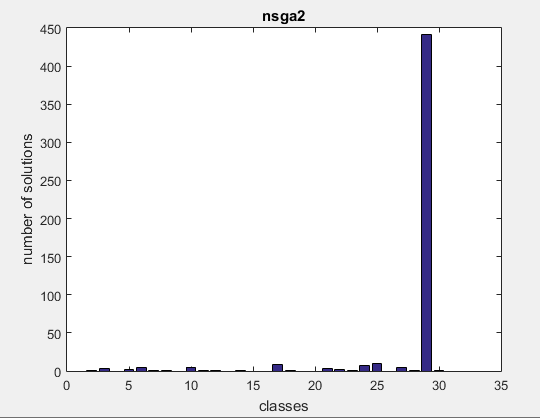












Conclusion:

1. Superiorety ratio of VLMOPSO over MOPSO for all 9 experiments is 44.44%
2. Superiorety ratio of MOPSO over VLMOPSO for all 9 experiments is 33.33%
3. Superiorety ratio of VLMOPSO over NSGA2 for all 9 experiments is 66.67%
4. Superiorety ratio of NSGA2 over VLMOPSO for all 9 experiments is 22.22%
5. There is need to make the max length bigger to make vlmopso dominates over other algorithms
6. There is need to tune some parameters like minimum number of particles in the class for vlmopso
7. In these experiments I added mutation for MOPSO, there is need to delete it because MOPSO is benchmark algorithm
8. In these algorithms I didn’t determine repository size in the last iteration in both MOPSO and VLMOPSO,

So you can find that pareto front size sometimes is bigger than population size

1. In vlmopso , we minimize random objective in each iterations, thus we may minimize one objective more than other, and this may be the reason behind that vlmopso some times go forword the classes that minimize one objective vs other, so we need to minimize the objectives **Alternately**

The End

Eng.Mohammad Sakka