Problem Set 9 **R08244002 Shin-Rong, Tsai**

- ## Simulation of 1D Real Scalar Field with Metropolis Algorithm

1. Result

Use the code from professor, and rewrite it as c language. Compile and run the code *scalar_field1D.c*. Then worked it through from $x = 0$ to $x = 100$.

Configuration:

$L = 1000 \quad N = 100 \quad \text{mass} = 0.01 \quad \text{max length of } \varphi: 100$

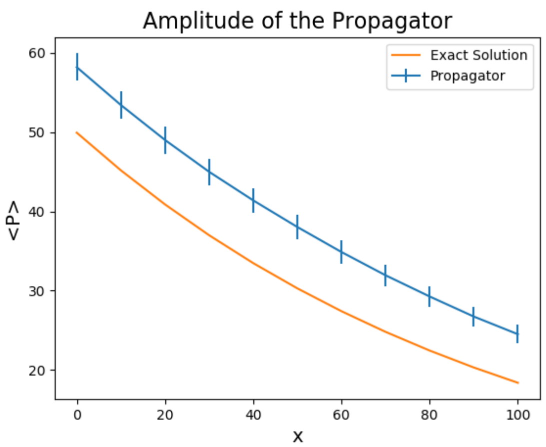
```
cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ gcc -o b.out scalar_field1D.c -lgsl

cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ ./b.out
Input output file name:
x10.txt
x10.txt
Enter Final point x :
10
10.000000
Choose the algorithm :
0:HMC
1:Metropolis
1
1
sweeps  mass          x-y           <S>           <P>           S_exact       P_exact       AC(%)
-----------------------------------------------------
2000    1.000000e-02  1.000000e+01  -4.776188e+00  4.587443e+01  -4.750289e+00  4.519188e+01  0.07
3000    1.000000e-02  1.000000e+01  -4.742235e+00  5.171328e+01  -4.750289e+00  4.519188e+01  0.07
4000    1.000000e-02  1.000000e+01  -4.766633e+00  6.202516e+01  -4.750289e+00  4.519188e+01  0.07
5000    1.000000e-02  1.000000e+01  -4.769614e+00  5.590108e+01  -4.750289e+00  4.519188e+01  0.07
6000    1.000000e-02  1.000000e+01  -4.794295e+00  5.287498e+01  -4.750289e+00  4.519188e+01  0.07
7000    1.000000e-02  1.000000e+01  -4.798391e+00  5.598553e+01  -4.750289e+00  4.519188e+01  0.07
8000    1.000000e-02  1.000000e+01  -4.780132e+00  5.502819e+01  -4.750289e+00  4.519188e+01  0.07
9000    1.000000e-02  1.000000e+01  -4.775076e+00  5.299315e+01  -4.750289e+00  4.519188e+01  0.07
10000   1.000000e-02  1.000000e+01  -4.773683e+00  5.118530e+01  -4.750289e+00  4.519188e+01  0.07
11000   1.000000e-02  1.000000e+01  -4.776336e+00  5.399916e+01  -4.750289e+00  4.519188e+01  0.07
```

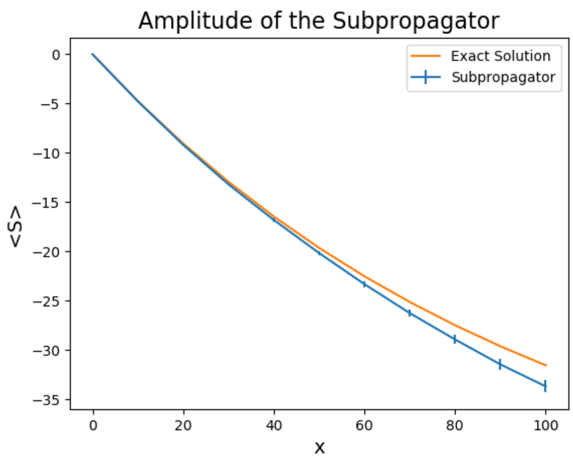
Then use the binning method with shell script *getBinning.sh* to find the error.

```
cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ gcc -o binning.out binning.c

cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ ./getBinning.sh
```

2. Discussion

Plot the result with python in file *Q1-plot.py*.





We can easily see that both of the result decay as propagate distance getting larger. In the sense of Quantum Mechanics, since there are more possible paths to choose from, the amplitude decays.

The subpropagator is more accurate since there is a datum point.

And also because of this, the error bars are getting bigger as the distance getting larger in subpropagator, the particles are harder to predict.

- ## Simulation of 1D Real Scalar Field with Hybrid Monte Carlo

1. Result

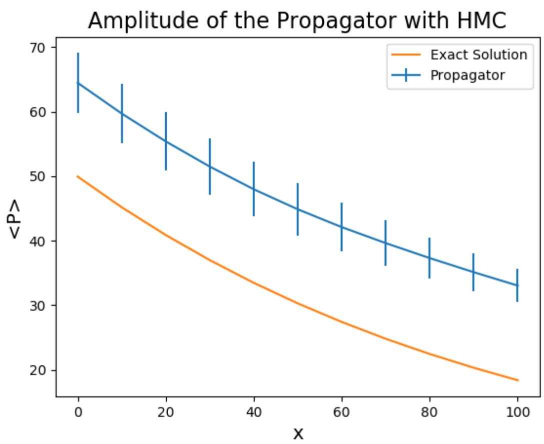
Use the same code as in previous question *scalar_field1D.c*. But choose the HMC algorithm. Use the same configuration as above too.

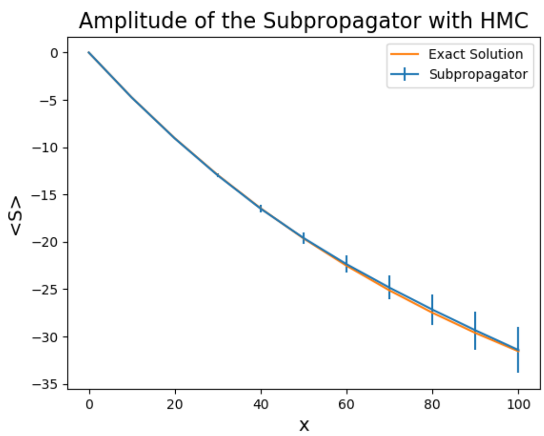
Then slightly change the shell script *getBinning.sh*, so that it directs to the folder of this algorithm.

```
cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ ./b.out
Input output file name:
x100.txt
x100.txt
Enter Final point x :
100
100.000000
Choose the algorithm :
0:HMC
1:Metropolis
0
0
sweeps  mass            x-y             <S>             <P>             S_exact         P_exact         AC(%)
------------------------------------------------------
2000    1.000000e-02    1.000000e+02    -2.068222e+01   2.807271e+01    -3.155648e+01   1.838569e+01    1.00
3000    1.000000e-02    1.000000e+02    -2.559887e+01   3.066335e+01    -3.155648e+01   1.838569e+01    1.00
4000    1.000000e-02    1.000000e+02    -3.256033e+01   3.869799e+01    -3.155648e+01   1.838569e+01    1.00
5000    1.000000e-02    1.000000e+02    -3.345292e+01   3.744730e+01    -3.155648e+01   1.838569e+01    1.00
6000    1.000000e-02    1.000000e+02    -3.323931e+01   3.886429e+01    -3.155648e+01   1.838569e+01    1.00
7000    1.000000e-02    1.000000e+02    -3.506603e+01   3.660924e+01    -3.155648e+01   1.838569e+01    1.00
8000    1.000000e-02    1.000000e+02    -3.410127e+01   3.310700e+01    -3.155648e+01   1.838569e+01    1.00
9000    1.000000e-02    1.000000e+02    -3.412778e+01   3.201116e+01    -3.155648e+01   1.838569e+01    1.00
10000   1.000000e-02    1.000000e+02    -3.297614e+01   3.199777e+01    -3.155648e+01   1.838569e+01    1.00
11000   1.000000e-02    1.000000e+02    -3.281333e+01   3.183774e+01    -3.155648e+01   1.838569e+01    1.00

cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ ./getBinning.sh
```

2. Discussion

Plot the result with python in file *Q2-plot.py*.





Compare with the Metropolis, HMC has a bigger error bar. The acceptance percentage is higher than Metropolis' which could lead to faster and more efficient computation, but these advantage is obtained in exchange of error bars.

We can see that the result of subpropagator matches the exact solution, since there is a datum point. The slope of the result directly from the simulation of propagators are alike.

The error bars are getting bigger in subpropagator, since their possible paths are getting much more, it is more unlikely to predict the particle.

- **1D Scalar Field with $\lambda\varphi^4$ Interaction**

1. Result

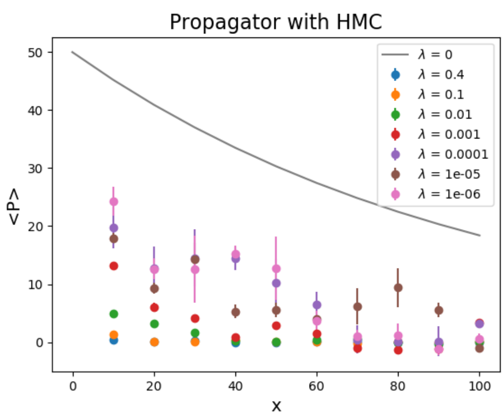
Add the interaction term to the lagrangian, compile and run the code *scalar1D_interaction.c*. And then run the shell script to analyze with binning method.

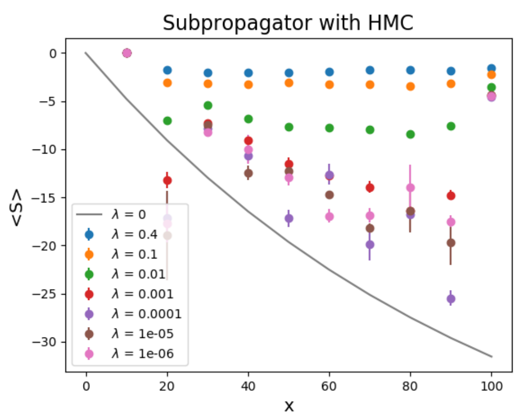
```
cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ gcc -o interaction.out scalar1D_interaction.c -lgsl

cindytsai@TURQUOISEA /cygdrive/d/GitHub/Computational_Physics/Assignment/ProblemSet_9
$ ./interaction.out
Enter the value of lambda :
0.4
0.400000
Choose the algorithm :
0:HMC
1:Metropolis
0
0
Enter range of the final point x we want to loop through to :
100
100.000000
sweeps  mass          x-y           <S>           <P>           S_exact       P_exact       AC(%)
-----------------------------------------------------------
2000    1.000000e-02  0.000000e+00  0.000000e+00  1.905911e+00  0.000000e+00  4.994217e+01  0.00
3000    1.000000e-02  0.000000e+00  0.000000e+00  1.916959e+00  0.000000e+00  4.994217e+01  0.00
4000    1.000000e-02  0.000000e+00  0.000000e+00  1.921811e+00  0.000000e+00  4.994217e+01  0.00
5000    1.000000e-02  0.000000e+00  0.000000e+00  1.924237e+00  0.000000e+00  4.994217e+01  0.00
6000    1.000000e-02  0.000000e+00  0.000000e+00  1.925692e+00  0.000000e+00  4.994217e+01  0.00
7000    1.000000e-02  0.000000e+00  0.000000e+00  1.926663e+00  0.000000e+00  4.994217e+01  0.00
```

2. Discussion

I use HMC algorithm to simulate the interaction. The acceptance percentage is lower, so the speed is lower.

After plot the binning result with python (in file *Q3-plot.py*), we have:





From which we can see that as $\lambda \to 0$, it is getting closer to the exact solution of the original $\lambda = 0$. So we can convince that it is indeed correct.