

ISTANBUL TECHNICAL UNIVERSITY

FACULTY OF SCIENCE AND LETTERS

Graduation Project



Machine Learning and Non-linear Schrödinger Equation

Hüseyin Talha Şenyaşa

Department : Physics Engineering

Student ID : 090120132

Advisor : Assoc. Prof. A. Levent Subaşı

FALL 2017

Summary

We train an artificial neural network to estimate the ground state energy of a one-dimensional Bose-Einstein condensate in harmonic trapping potential. Such a system can be described by the solution of a non-linear Schrödinger equation also called a Gross-Pitaevskii equation. We also use the method for the inverse problem of predicting the non-linearity parameter using the ground state density profile for a given harmonic trapping potential.

Contents

1	Introduction and Motivation	1
2	Gross Pitaevskii Equation	1
2.1	General information about GPE	1
2.2	Why and how nonlinearity is introduced.	1
2.3	Physical and mathematical interpretation of interaction parameter. (phy: attractive, repulsive math:dominance of the terms) . . .	1
2.4	Stationary form.	1
2.5	Potential, kinetic and interaction energy expressions.	1
2.6	Reduction of dimension.	1
2.7	Analytic solution and approximation.	1
2.8	Numeric Solution and Dataset Generation	1
2.8.1	Potential types (with analytic forms etc.)	1
2.8.2	Scaling	2
2.8.3	Brief info about imaginary time evolution. (detailed in APPENDIX)	4
2.8.4	XMDS framerwork and other programs.	4
2.8.5	Potential generation	4
2.8.6	Random potential generations with different method. (Reason)	5
2.8.7	Boundaries. (Table)	6
2.8.8	Convergence (detailed in APPENDIX)	6
2.8.9	Dataset generation. (Total number of examples etc)	6
2.9	Dataset Features	6
2.9.1	Energy distribution	6
3	Machine Learning	6
3.1	Network architecture	6
3.2	Training	7
3.3	Results	7
4	Inverse Problem	7
5	Conclusion and Discussion	8
5.1	Conclusion	8
5.2	Discussion	8
5.3	Effects of random potential generation method	8
5.4	Are there problems in low and high energies compared to the mean	8
5.5	Inverse problem	8

1 Introduction and Motivation

Machine learning.

General usage area.

ML in physics and Physics in ML.

ML&SE article.

Ours difference.

2 Gross Pitaevskii Equation

2.1 General information about GPE

2.2 Why and how nonlinearity is introduced.

2.3 Physical and mathematical interpretation of interaction parameter. (phy: attractive, repulsive math:dominance of the terms)

2.4 Stationary form.

2.5 Potential, kinetic and interaction energy expressions.

2.6 Reduction of dimension.

2.7 Analytic solution and approximation.

2.8 Numeric Solution and Dataset Generation

2.8.1 Potential types (with analytic forms etc.)

We use **six** different types of potential in our study. The first three potential which have analytic form given in Table 1. are 1D version of potentials studied in [1]. We implement two more random potential generator to study effect of generation process. The generation processes of the random potentials are described in Section 2.8.6.

Table 1: Potential Types and Analytic Forms

Potential Type	Analytic Form
Infinite Well	$V(x) = \begin{cases} 0 & \text{if } x_l < x < x_r \\ \infty & \text{if } otherwise \end{cases}$
Harmonic	$V(x) = \frac{1}{2}m\omega^2x^2$
DI Gaussian	$V(x) = -A_1\exp(\frac{(x-\mu_1)^2}{s_1^2}) - A_2\exp(\frac{(x-\mu_2)^2}{s_2^2})$

Table 2: Parameters

Potentials	Parameters	Min	Max
Infinite Well	x_l	-9	4.5
	x_r	-4.5	9
	$x_l - x_r$	1	8
Harmonic	ω	0.01	4
	x_0	-5	5
DI Gaussian	A_1, A_2	1	10
	μ_1, μ_2	-5	5
	s_1, s_2	0.5	4

2.8.2 Scaling

The scaling of GPE is generally done according to potential type and there are more than one scaling conventions **REFS**. In our study, we use a more general scaling to investigate how different scalings affect the precision. To do that, we compare numerical solutions by representing same physical system with different scaling coefficients in both framework. Then we compare these results first internally and then we do a cross check.

We also compare numerical solutions of two different solvers implemented in different frameworks to show that the numerical solutions are **consistent**.

GPE is given as,

$$\frac{-\hbar^2}{2m} \frac{d^2\psi}{dz^2} + V(z)\psi + g|\psi|^2\psi = \mu\psi \quad (1)$$

First we define dimensionless potential, and then we make the length dimensionless,

$$\tilde{V}(z) \equiv \frac{V(z)}{\gamma E_0}$$

$$\tilde{z} \equiv \frac{z}{\beta L}$$

The potentials are going to be generated according to the length transformation, thus; we can transform potential expression implicitly.

$$\tilde{V}(z) \rightarrow \tilde{V}(\tilde{z})$$

If these transformations are plugged into the Eq. (1) it becomes,

$$\frac{-\hbar^2}{2m\gamma E_0} \frac{1}{\beta^2 L^2} \frac{d^2\psi}{d\tilde{z}^2} + \tilde{V}(\tilde{z})\psi + \frac{g}{\gamma E_0} |\psi|^2\psi = \frac{\mu}{\gamma E_0} \psi \quad (2)$$

To obtain final form, we define dimensionless energy, wave function and interaction parameter respectively.

$$\tilde{\mu} \equiv \frac{\mu}{\gamma E_0}$$

$$\tilde{\psi} \equiv \psi \sqrt{\frac{\beta L}{N}}$$

$$\tilde{g} \equiv \frac{gN}{\gamma E_0 \beta L}$$

To control scaling coefficients we set the coefficient of the kinetic term to an arbitrary positive real number α

$$\frac{\hbar^2}{2m\gamma E_0} \frac{1}{\beta^2 L^2} = \alpha,$$

and set $E_0 = \hbar^2/2m$. Now the scaling of GPE can be controlled by α and β only.

$$-\alpha \frac{d^2\tilde{\psi}}{d\tilde{z}^2} + \tilde{V}(\tilde{z})\tilde{\psi} + \tilde{g}|\tilde{\psi}|^2\tilde{\psi} = \tilde{\mu}\tilde{\psi} \quad (3)$$

We are going to change these scaling coefficients and see their effects by comparing results. An example of setting the scale coefficients is shown in appendix A.

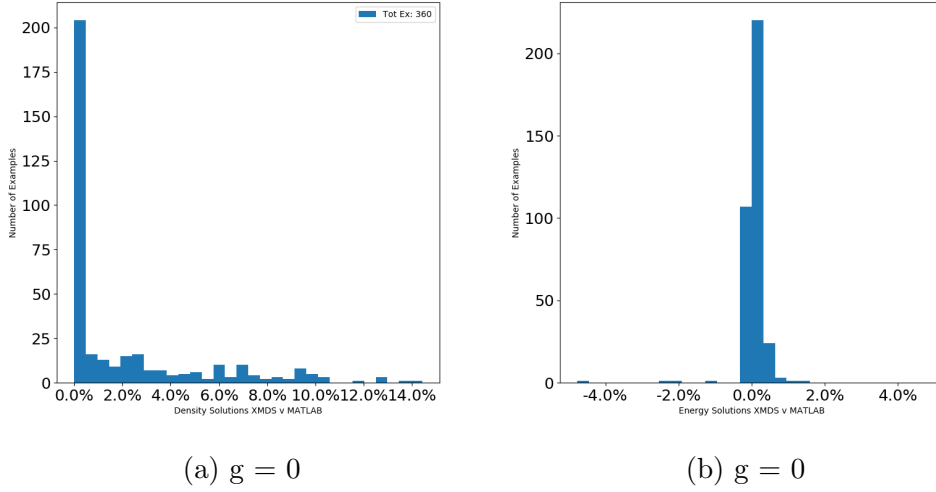


Figure 1

2.8.3 Brief info about imaginary time evolution. (detailed in APPENDIX)

2.8.4 XMDs framework and other programs.

2.8.5 Potential generation

We create an envelope function as a mask with the following expression,

$$\tanh(\beta(x + x_l)) + \tanh(\beta(x + x_r)) \quad (4)$$

where x_l and x_r bounds are given in the **table**. We apply this mask to all potentials except harmonic to make sure that all potential goes to maximum value at numerical infinity.

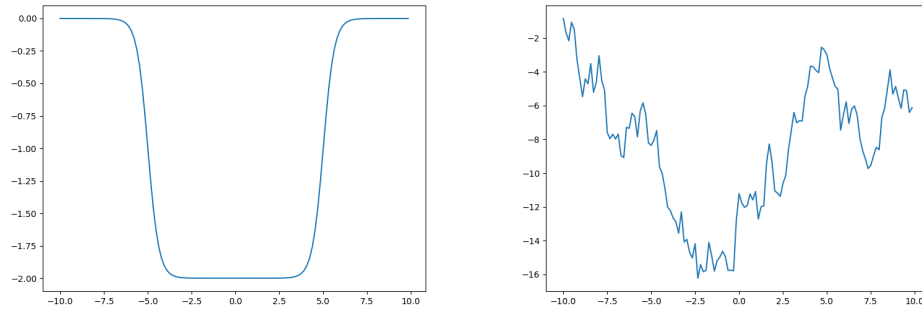


Figure 2: Envelope Potential and Random Potential Before Process

The two potential in Figure 2 are combined and a gaussian filter applied.

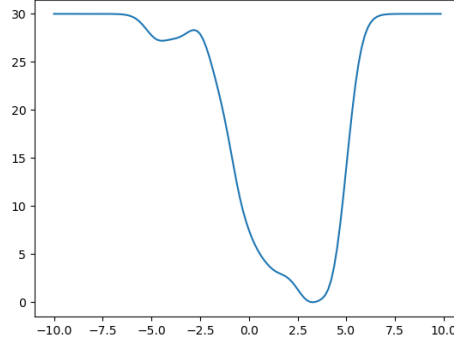


Figure 3

2.8.6 Random potential generations with different method. (Reason)

To be able to observe the effect of random potential generation to the results we used three different random potential generation algorithms. The first one is random walk with random step size such that the first value of the potential array is initialized with a random number. After that, another random number is added to this value to obtain the next element of the array and so on. The distribution of the random numbers is gaussian in this process. The resultant array is not guaranteed to be smooth. We applied a gaussian filter with a random sigma value to the potential array to make it smooth.

Algorithm 1 RandomPotential1

```

1: procedure RANDOMPOTENTIAL1
2:   Points = GaussianDistributedRandomPoints()
3:   Len = Length(Points)
4:   Potential[0] = Points[0]
5:   for i = 0 to Len − 1 do
6:     Potential[i + 1] = Pot[i] + Points[i]
7:   Potential = GaussianFilter(Potential,  $\sigma$ )

```

The second one is summation of sines and cosines with random coefficients. Firstly, the number of terms are determined, after that each coefficient of the terms are assigned and summed iteratively. The resultant sum is potential and a gaussian filter is applied.

Algorithm 2 RandomPotential2

```
1: procedure RANDOMPOTENTIAL2
2:    $Nterms = RandomInteger(1, 100)$ 
3:   for  $i = 0$  to  $Nterms$  do
4:      $A = GaussianDistributedRandomNumber()$ 
5:      $B = GaussianDistributedRandomNumber()$ 
6:      $n_1 = GaussianDistributedRandomNumber() * \sigma * \pi / width$ 
7:      $n_2 = GaussianDistributedRandomNumber() * \sigma * \pi / width$ 
8:      $Potential += A \sin(n_1 x) + B \cos(n_2 x)$ 
9:    $Potential = GaussianFilter(Potential, \sigma)$ 
```

We use slight modified version of the algorithm described in [1]. Our modification occurs in the masking part. In this part we use our envelope function rather than convex hull mask.

Algorithm 3 RandomPotential3

```
1: procedure RANDOMPOTENTIAL3
2:    $ScaleFactor = 8$ 
3:   for  $i = 0$  to  $NumberOfPoints/ScaleFactor$  do
4:      $BinaryGrid[i] = RandomInteger(0, 1)$ 
5:    $BinaryGrid = RepeatElements(BinaryGrid, ScaleFactor)$ 
6:   for  $i = 0$  to  $NumberOfPoints/(ScaleFactor * 2)$  do
7:      $BinaryGrid2[i] = RandomInteger(0, 1)$ 
8:    $BinaryGrid2 = RepeatElements(BinaryGrid, ScaleFactor)$ 
9:    $Padding = Zeros((Length(BinaryGrid) - Length(BinaryGrid2))/2)$ 
10:   $BinaryGrid2 = Concatenate(Padding, BinaryGrid2, Padding)$ 
11:   $Potential = BinaryGrid - BinaryGrid2$ 
12:   $Potential = GaussianFilter(Potential, \sigma)$ 
```

2.8.7 Boundaries. (Table)

2.8.8 Convergence (detailed in APPENDIX)

2.8.9 Dataset generation. (Total number of examples etc)

2.9 Dataset Features

2.9.1 Energy distribution

3 Machine Learning

3.1 Network architecture

Architecture of the network.

A general figure like in the ML&SE article that describes the work done.

Another figure about internals of the network such as number of layers, how interaction parameter is introduced to the network etc.
Hyperparameters.

3.2 Training

Detailed info about dataset (energy distribution etc).

Indicate that if there is any method to increase the number of examples in low and high energy values.

3.3 Results

4 Inverse Problem

5 Conclusion and Discussion

5.1 Conclusion

5.2 Discussion

5.3 Effects of random potential generation method

5.4 Are there problems in low and high energies compared to the mean

5.5 Inverse problem

References

- [1] K. Mills, M. Spanner, and I. Tamblyn, “Deep learning and the schrödinger equation,” *Physical Review A*, vol. 96, no. 4, p. 042113, 2017.

A APPENDIX A

Obtaining harmonic trap potential scaling from Eq. (3)

$$V(z) \rightarrow V(\tilde{z}) \rightarrow \tilde{V}(\tilde{z}) \quad (5)$$

$$\tilde{V}(z) \equiv \frac{V(z)}{\gamma E_0} \quad (6)$$

$$\tilde{z} \equiv \frac{z}{\beta L} \quad (7)$$

$$V(z) = \frac{1}{2} m \omega^2 (z - z_0)^2 \quad (8)$$

$$V(\tilde{z}) = \frac{1}{2} m \omega^2 \beta^2 L^2 (\tilde{z} - \tilde{z}_0)^2 \quad (9)$$

$$\tilde{V}(\tilde{z}) = \frac{1}{2} m \omega^2 \frac{\beta^2 L^2}{\gamma E_0} (\tilde{z} - \tilde{z}_0)^2 \quad (10)$$

The coefficient of this equation must be dimensionless, therefore;

$$\frac{1}{2} m \omega^2 \frac{\beta^2 L^2}{\gamma E_0} = C \quad (11)$$

Where C is a positive constant. We know that $E_0 = \frac{\hbar^2}{2m}$ and the definition of α is given as,

$$\frac{\hbar^2}{2m\gamma E_0} \frac{1}{\beta^2 L^2} = \alpha \quad (12)$$

thus;

$$\frac{1}{\gamma} = \alpha \beta^2 L^2 \quad (13)$$

if we plug Eq. (13) in to Eq. (11), then equation becomes,

$$\frac{m^2 \omega^2}{\hbar^2} \alpha \beta^4 L^4 = C \quad (14)$$

in the case of expressing same physical system, m and ω must be constant. In

that case $\alpha = C(\beta L)^{-4}$.

In this case, α becomes

$$\alpha = \frac{1}{2} \left(\frac{\hbar\omega}{\gamma E_0} \right)^2$$

Conventionally, α is set to 1/2, therefore;

$$\hbar\omega = \gamma E_0$$

$$\beta L = \sqrt{\frac{\hbar}{m\omega}}$$

βL is generally defined as harmonic oscillator length ℓ

$$\ell = \sqrt{\frac{\hbar}{m\omega}}$$

$$\tilde{\mu} = \frac{\mu}{\hbar\omega}$$

$$\tilde{g} = \frac{g}{\hbar\omega\ell}$$

Finally, dimensionless GPE scaled for harmonic trapping potential can be written as,

$$\tilde{\mu}\tilde{\psi} = -\frac{1}{2}\frac{d^2\tilde{\psi}}{d\tilde{z}^2} + \frac{1}{2}\tilde{z}^2\tilde{\psi} + \tilde{g}|\tilde{\psi}|^2\tilde{\psi}$$