Report on Project II

Bin Fan

March 11, 2011

Abstract

We implemented a program which applies the shooting method to finite quantum well. The algorithm has been implemented on a very primitive level which can only perform the computation on uniform meshgrids with the Runge-Kutta fourth order method. In this report, we first describe our quantum well problem, the Shooting method, and our implementation details. After that, we present our solutions and the analyses.

1 Software Manual

Problem Description We are presented with a simplified quantum well problem. The potential is

$$V(x) = \begin{cases} -V_0 & \text{if } x \geq \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Also we have the simplified shcrondinger equation, in which we put m, $\hbar$ all to 1;

$$\frac{\partial^2 \Psi}{\partial x^2} = (E - V)\Psi$$

Based on the two above equations, we can find all the energy levels for this boundary problem.

Shooting method The shooting method derives from the idea that we can solve a boundary value problem by reducing it to the solution of an initial value problem

We can both use the Runge-kutta and Eular algorithm to integrate this second order initial function. After integration, the $\Psi(x)$ will arive at the other side of the boundary, we just need to match its value with our preset boundary value, and judge whether the initial value if of interest.

Bisection root finding Since we are capable of knowing whether the $\Psi(x)$ will go upward or downward as x increases, what we need to do is just to find the root of the equation that set $\Psi(x, E)$ go to 0 in the infinity.

We know how to do this by several ways, however, the easiest one is to search the root by Bisection Method, which repeatedly bisects an interval then selects a subinterval in which the root lie in by judging the sign of function on the two edge of interval.

We use a recursion function to perform this task. The program will automatically output all energy levels in even or odd parity.

Program Usage Type ./shoot in the directory will envoke the program, but user can also specify algorithm, maximum energy, increment step, even or odd parity, plot or not by feeding command line arguments.

-h *[increment step]* : specify the increment step for integrate Ψ

-p : tell the program just compute the $\Psi(x)$ and show it (used acompany with -e), otherwise the program will skip this and search for energy levels

-r : use Runge-kutta algorithm, default using Eular algorithm

-o : odd parity, default even parity

-e *[energy]* : set E, used acompany with -p option

Example:

./shoot -p -e -0.088 invoke the program to integrate $\Psi(x)$ with the energy of $-0.088V_0$

./shoot ask the program to search for all energy levels

2 Final Result

We use Runge-Kutta integrator, step size is 0.0001, with odd parity. The program finds four eigen energy values by repeatedly bisection. In the next part, we show the two adjacent energy levels for each eigen energy value. Each page show two images corresponding to one eigen energy level.

Figure 1: for the energy level of 0.088497

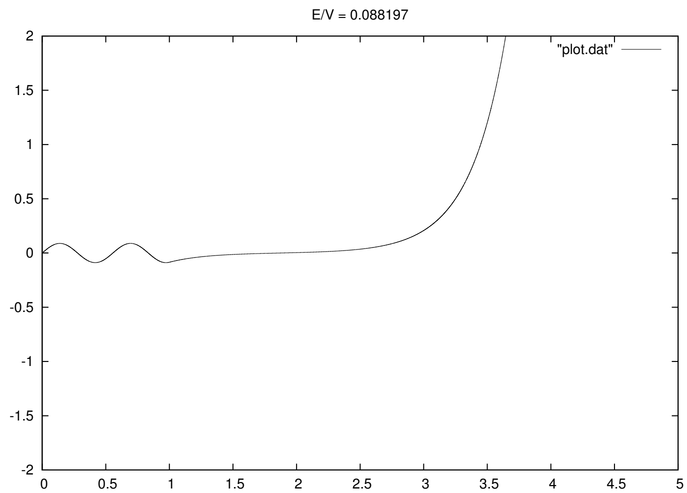


Figure 2: for the energy level of 0.88497

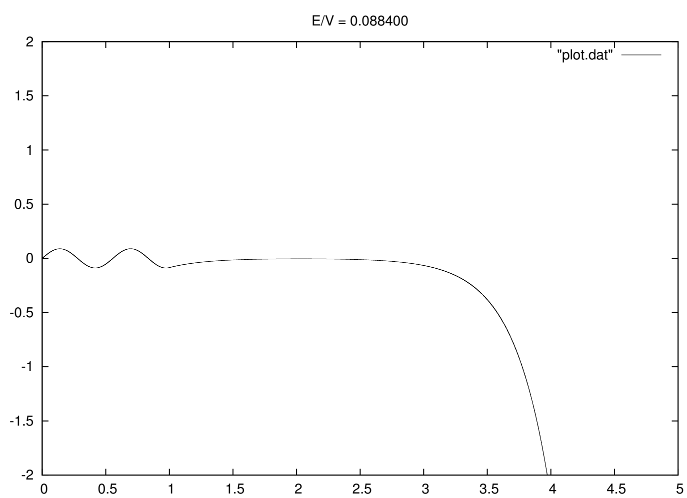


Figure 3: for the energy level of 0.470697

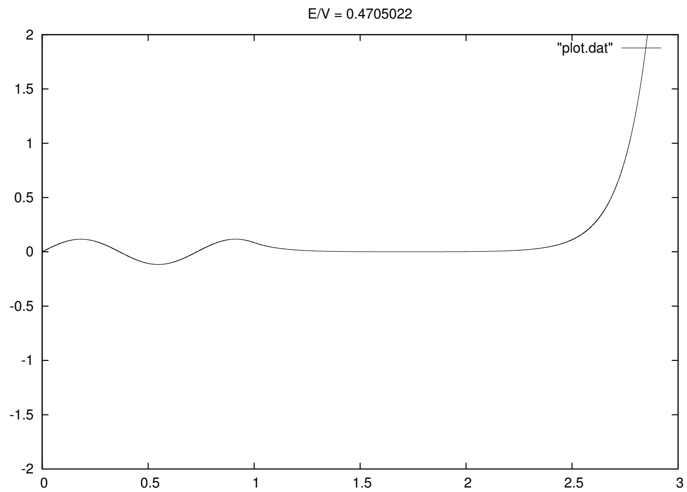


Figure 4: for the energy level of 0.470697

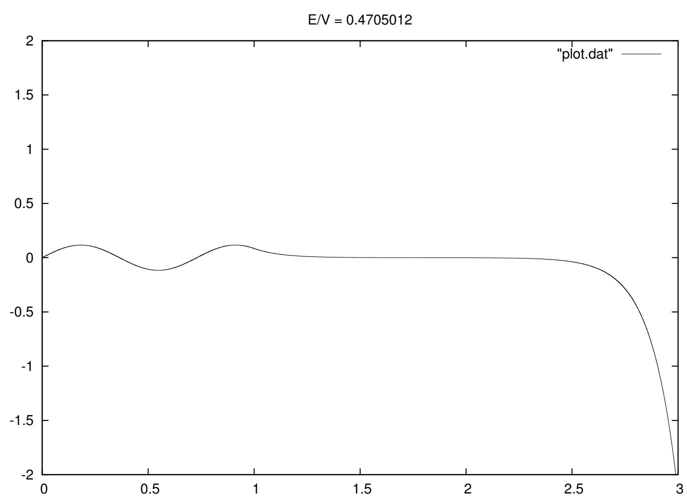


Figure 5: for the energy level of 0.762097

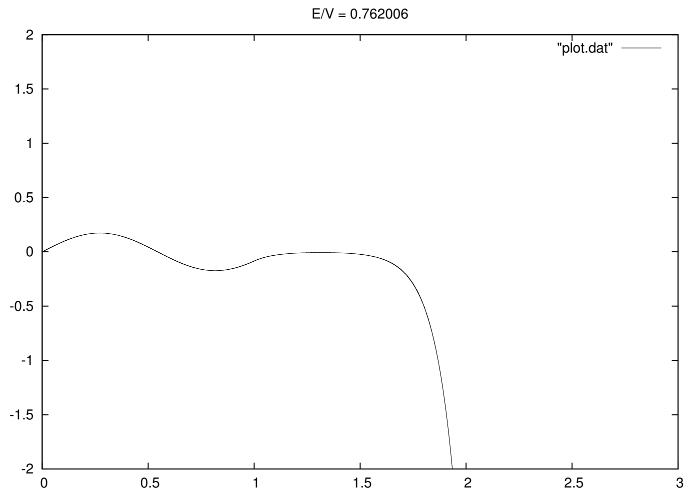


Figure 6: for the energy level of 0.762097

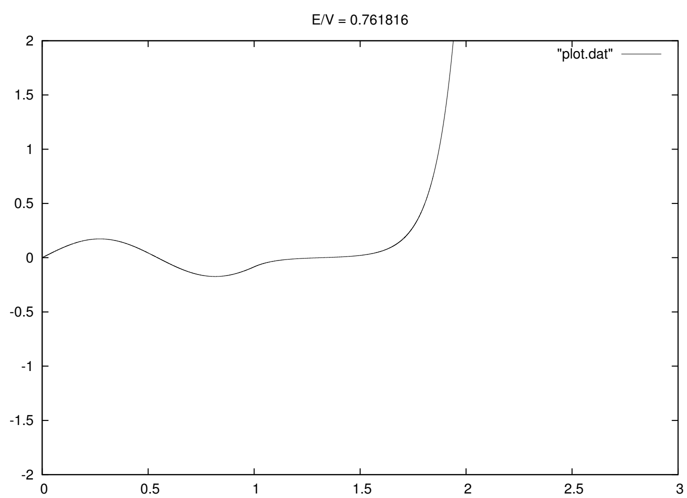


Figure 7: for the energy level of 0.94029

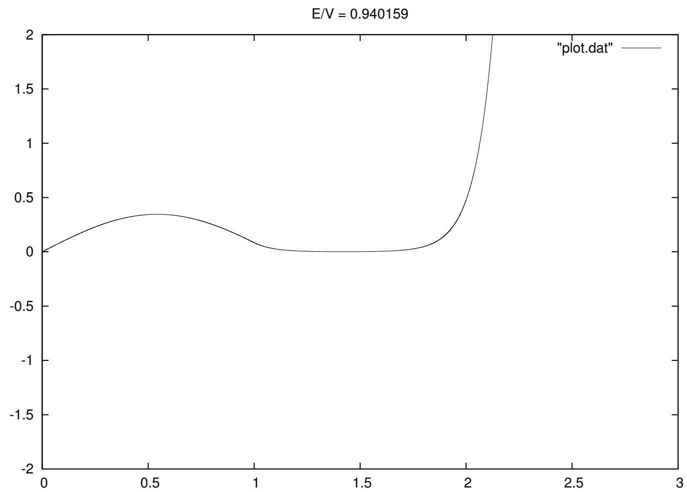
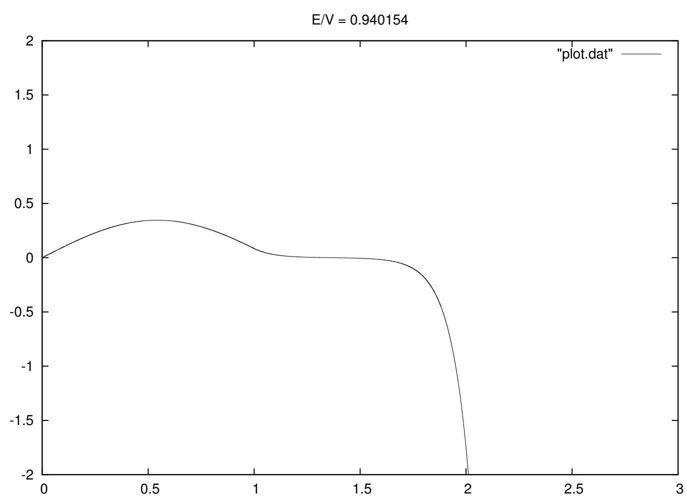


Figure 8: for the energy level of 0.94029

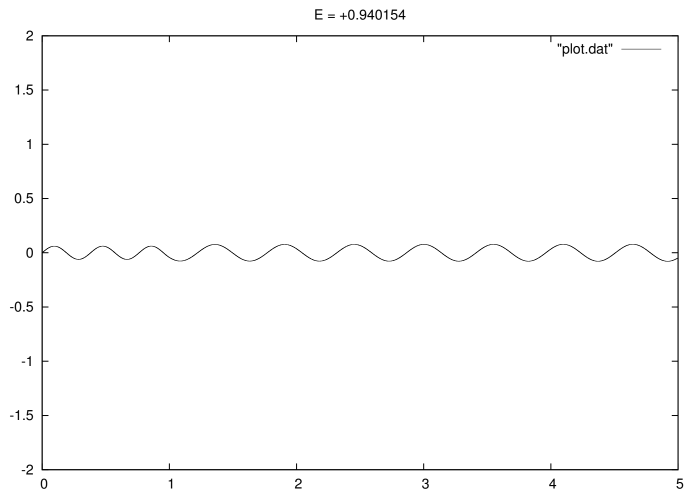


After these, we see that with the energy E more close to the bottom, the property of $\Psi(x)$ is more assympototic around the energy level. Also, we can see that the number of nodes between 0 and 1 are decided by energy level n

Next we show several special energy levels.

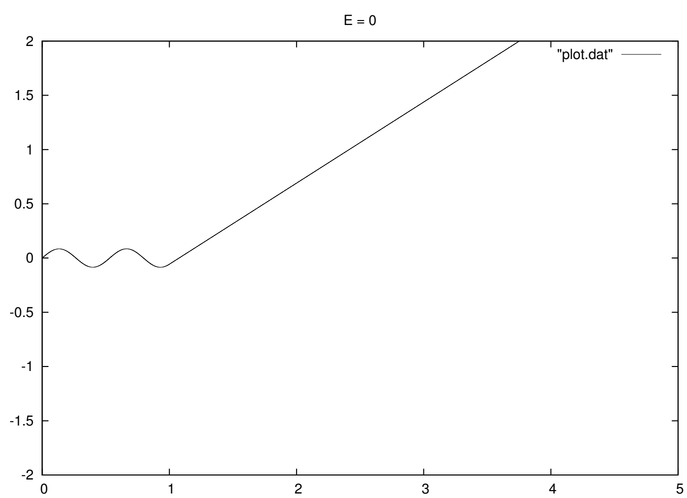
3 Special Energy

Figure 9: $E > 0$ Particle out of the well



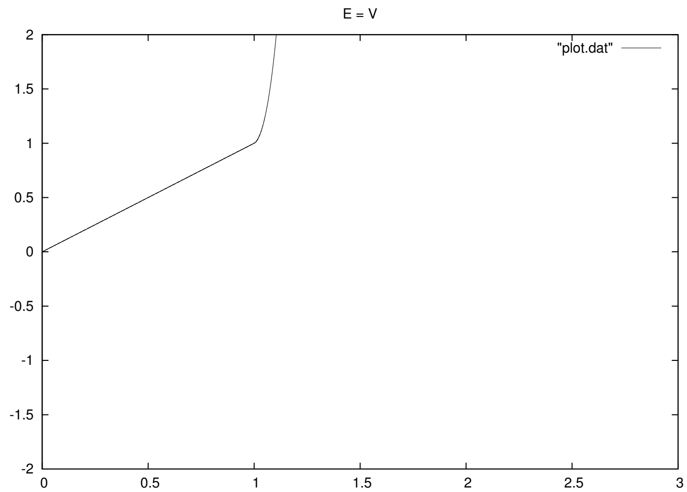
Even though it is out of the well, we can notice that between x=0 and x=1, the particle still “feels” the well potential.

Figure 10: $E = 0$ Particle just on the top of the well



Particle here are extremely sensitive to the energy

Figure 11: $E = V$ Particle at the bottom of the well



Even between x=0 and x=1, the particle is exploding.