The Shooting Method

In this illustration I show how to solve the harmonic oscillator energy levels numerically. We begin by defining the potential energy function. Since computers don't use units I have selected units where the mass and spring constant, and hbar, all have numerical parts equal to one. (So the expected energy levels \((n+1/2)\hbar\omega\) are just \(1/2, 3/2, \ldots\))

\[
VHO := x \rightarrow \frac{1}{2} x^2
\]

Next I define the Schrodinger equation for energy level \(ET\)

\[
\text{Schr} := -\frac{d^2}{dx^2} \psi(x) + \left( \frac{1}{2} x^2 - ET \right) \psi(x) = 0
\]

Now I need initial conditions to solve this equation. It's second order, so I need two conditions, one on \(\psi\) and one on \(\psi'\).

It turns out (ask me if you're curious) that half the solutions will be symmetric functions; that is, \(\psi(-x) = \psi(x)\). This implies that \(\psi'(0) = 0\).

I'm not interested in the normalization of \(\psi\), so any symmetric solution can be scaled until \(\psi(0) = 1\). So the initial conditions for the symmetric solutions are

\[
\text{ICsym} := \psi(0) = 1, \psi'(0) = 0
\]

I need a certain package:

\[
\text{with(plots):}
\]

Now I call a long gobbledygook (which you don't need to understand) to solve the equation for a chosen value of energy and plot the answer. I just happen to know the exact solution to this special problem, so I put in the right \(ET=0.5\). Note the colon at the end suppresses the output.

\[
\text{P1:=odeplot(dsolve(\{subs(ET=0.5,Schr),ICsym\},\psi(x),numeric),[x,\psi(x)],}
\]

\[
-1 -
\]

The solution is in blue below. For reference I'll also plot the energy (black) and a line at ET (red). I scaled them both by a factor of 3 so they'd fit nicely. The thing to notice is that these lines cross at \( x=1 \), so any nonzero \( \psi \) beyond that point is classically forbidden.

So there's lots of probability outside the classical turning point -- tunneling is possible. Next I show what happens if our guess for the energy value is a bit high:

> odeplot(dsolve({subs(ET=0.55,Schr),ICsym},psi(x),numeric),[x,psi(x)],0..3,-2..2,title="0.55");

Error, (in dsolve) invalid arguments
The solution rapidly goes to minus infinity! And if we guess a value too low for ET:

```maple
> odeplot(dsolve({subs(ET=0.46,Schr),ICsym},psi(x),numeric),[x,psi(x)],
0..3,-2..2,title=`0.46`);
Error, (in dsolve) invalid arguments
```

Finally here is a much larger value for ET (remember, we only get the symmetric solutions with this boundary condition). Again I plot the potential and a straight line at ET to show that the solution passes from wavy to damped as it crosses into the classically-forbidden zone:

```maple
> P1:=odeplot(dsolve({subs(ET=8.5,Schr),ICsym},psi(x),numeric),[x,psi(x)],
0..6,-2..4,title=`E=8.5`):
> P2:=plot({VHO(x)/5,8.5/5},x=0..6,y=-2..4): display({P1,P2});
```

> 

You may also notice that the amplitude of the wavy part increases towards the edge of the classically forbidden region. That is, the probability to be found there increases. That makes sense -- classically the particle pauses at its turning points, so it spends more time there and is more likely to be found there at any instant!