Scattering States in a 2D Quantum Dot Embedded in a Waveguide

Michael Trott

The eigenfunctions belonging to the continuous spectrum of a circular quantum dot connected to two infinite strip-like wires are calculated using the mode-matching technique. The corresponding quantum-mechanical probability densities, currents, and transmission and reflection coefficients are calculated and visualized.

Introduction

In the 9:3 Corner, I discussed the sound of a fractal drum. More technically speaking, I calculated selected point spectrum eigenvalues and eigenfunctions of the Helmholtz equation of a complicated, bounded 2D domain with homogeneous Dirichlet boundary conditions. In this Corner, I will treat a problem similar in spirit—I will calculate some eigenfunctions of the continuous spectrum of an unbounded 2D domain with homogeneous Dirichlet boundary conditions. More concretely, we will study the transmission of a wave through a 2D quantum dot in the stationary regime. Here is a sketch of the domain $\Omega$ under consideration: (the $x$-direction runs to the left and the $y$-direction runs upwards).
\[\text{In[1]}=\text{Module}\{R=10, L=5, \phi = \text{ArcSin}[(L/2)/R]\};\]

\[\text{Show}\left[\text{Graphics}\left[\text{(* left and right arms and center disk *)}\right.\right.\right.\]
\[\left.\left.\left.\text{GrayLevel[0.8], Disk[\{0, 0\}, R],}\right.\right.\right.\]
\[\left.\left.\left.\text{Rectangle[\{-2 R, -L/2\}, \{2 R, L/2\}]}\right.\right.\right.\]
\[\left.\left.\left.\text{(* boundaries with homogeneous Dirichlet boundary conditions *)}\right.\right.\right.\]
\[\left.\left.\left.\text{GrayLevel[0], Thickness[0.01], Circle[\{0, 0\}, R, \{\phi, Pi - \phi\}],}\right.\right.\right.\]
\[\left.\left.\left.\text{Circle[\{0, 0\}, R, \{Pi + \phi, 2 Pi - \phi\}],}\right.\right.\right.\]
\[\left.\left.\left.\text{Line[\{-2 R, +L/2\}, \{-R Cos[\phi], +L/2\}],}\right.\right.\right.\]
\[\left.\left.\left.\text{Line[\{2 R, +L/2\}, \{R Cos[\phi], +L/2\}],}\right.\right.\right.\]
\[\left.\left.\left.\text{Line[\{-2 R, -L/2\}, \{-R Cos[\phi], -L/2\}],}\right.\right.\right.\]
\[\left.\left.\left.\text{Line[\{2 R, -L/2\}, \{R Cos[\phi], -L/2\}]}\right.\right.\right.\]
\[\left.\left.\left.\text{]}\right.\right.\right.\]
\[\text{PlotRange \rightarrow All, AspectRatio \rightarrow Automatic}\right]\]

The arms have lateral width \(L\) and infinite horizontal extension, and the center disk has radius \(R\). To make calculations simpler, we take the arms to extend up to the circular boundary of the center disk.

We will solve the elliptic eigenvalue problem \(-\Delta \psi(r) = \mathcal{E} \psi(r)\) with the boundary condition \(\psi(r)|_{\partial \Omega} = 0\). We will assume an incoming wave from the left with unit strength of the form \(\exp(ikx)\chi(y)\). We will use the mode-matching technique to solve this Helmholtz equation. That is, we will construct within neighboring regions locally parametrized complete families of solutions and join them smoothly along certain curves by fixing the values of the parameters. We will naturally choose the left arm, the center disk, and the right arm for the regions.

A practical realization of such structures are quantum dots [1–8]. The Helmholtz equation in this case is identical to the time-independent Schrödinger equation with energy \(\mathcal{E}\). We will use this model as the physical interpretation in our calculations.

Because the domain \(\Omega\) is symmetrical, we can construct eigenfunctions with a definite parity with respect to reflections in the \(x\)-axis. We will always consider symmetric states. The construction of antisymmetric states is completely analogous.
Inside the left arm, we can write the general solution of the Helmholtz equation in the form

\[ \psi(x, y) = e^{ik_1 x} + \sum_{m=1}^{\infty} \alpha_m e^{-ik_m x} \cos \left( \frac{(2m-1)\pi y}{L} \right) \approx \]

\[ e^{ik_1 x} + \sum_{m=1}^{p} \alpha_m e^{-ik_m x} \cos \left( \frac{(2m-1)\pi y}{L} \right) = \lim_{x \to -\infty} \left( e^{ik_1 x} + R e^{-ik_1 x} \cos \left( \frac{\pi y}{L} \right) \right). \]

Here the inhomogeneous term has strength 1. We truncate the infinite sum because the contributions of the higher states decay exponentially with \(m\) for energies in the range \((\pi/L)^2 < E < (3\pi/L)^2\) (the lowest sub-band in the \(y\)-direction is occupied). The lateral wave number \(k_m\) is, for a given energy \(E\), defined through \(E = k_m^2 + ((2m-1)\pi/L)^2\). The first term in the expansion contains the reflection coefficient \(R\). The form of the \(y\)-dependent part automatically ensures the fulfillment of the Dirichlet boundary conditions in the lower and upper boundaries of the left arm.

Similarly, in the right arm, we use the expansion

\[ \psi(x, y) = \sum_{m=1}^{\infty} \gamma_m e^{ik_m x} \cos \left( \frac{(2m-1)\pi y}{L} \right) \approx \]

\[ \sum_{m=1}^{q} \gamma_m e^{ik_m x} \cos \left( \frac{(2m-1)\pi y}{L} \right) = \lim_{x \to -\infty} \left( e^{ik_1 x} + T e^{ik_1 x} \cos \left( \frac{\pi y}{L} \right) \right). \]

Again, we truncate the infinite sum. This time, the first term in the expansion contains the transmission coefficient \(T\) and \(k_m\) is defined as in the left arm.

Inside the center disk, we use a polar coordinate system. In polar coordinates, any solution of the Helmholtz equation can be written in the form

\[ \psi(r, \varphi) = \sum_{m=0}^{\infty} \beta_m \cos(m \varphi) J_m(\sqrt{E} \ r) \approx \sum_{m=0}^{q} \beta_m \cos(m \varphi) J_m(\sqrt{E} \ r). \]

Here the \(J_m(\xi)\) are Bessel functions.

The \(p + q + (\rho + 1)\) unknown coefficients \(\alpha_k\), \(\beta_k\), and \(\gamma_k\) are determined through the continuity of \(\psi(x, y)\), the first derivative of \(\psi(x, y)\) along curves joining the left and right arms with the center disk, and the fulfillment of the Dirichlet boundary condition along the arcs between the arms and the center disk. We will unite the three sets of coefficients \(\alpha_k\), \(\beta_k\), and \(\gamma_k\) into the \(p + q + (\rho + 1)\) coefficients \(c_k\) and use the following convention:

\[ c_1, \ldots, c_p \equiv \alpha_1, \ldots, \alpha_p \text{ (left arm)} \]
\[ c_{p+1}, \ldots, c_{p+q} \equiv \gamma_1, \ldots, \gamma_p \text{ (right arm)} \]
\[ c_{p+q+1}, \ldots, c_{p+q+\rho+1} \equiv \beta_0, \ldots, \beta_{\rho+1} \text{ (center disk)}. \]
Calculate Fourier Coefficients

To determine the coefficients $c_k$, we must construct equations encoding the continuity of the solution and its first derivative. As the joining curves, we will use the parts of the boundary of the center disk that intersects both arms. Along these arcs, we demand that the first $d$ Fourier coefficients of the difference between the solution ansatz of the arms and of the disk vanish.

We denote by $\phi$ the angle from the positive $x$-axis to the intersection of the center disk with the upper boundary of the right arm.

We use these global parameters: the energy $\mathcal{E}$, the number of states in the three regions $p$, $o + 1$, and $q$, the radius and width $R$ and $L$, and $\phi$.

To abbreviate the following inputs, we define these solution ingredients: a left running mode, a right running mode, and a mode for the center disk. Though we restrict ourselves to solutions symmetric with respect to the $x$-axis, the antisymmetric states can be obtained by replacing cos with sin in the fundamental modes.

Here are the basis functions for the expansion in the three regions.

On the left arc, we multiply with the first of the functions $c_L$, $c_R$, $w$ that is of the form $\cos \sin \frac{\left( n(\phi - \pi) / \phi \right)}{\pi / 2}$ (they form an overcomplete system but cover the arc in a homogeneous manner). Later, we will use some of the Fourier coefficients (the cos-modes with odd $n$ that are complete for the symmetric states we are looking for) and demand that they vanish.

Here are these Fourier modes. Because the solution of the problem has to fulfill homogeneous Dirichlet boundary conditions along the boundaries of the center disk, we will later use the cos-modes that vanish at the endpoints of these arcs.
The function cond[\text{leftRight}, \alpha, \mu, \{cs, n\}] represents the continuity of the \(\mu\)th mode on the left or right arm arc of the \(n\)th derivative (\(\alpha = 0\) or \(\alpha = 1\) in the following) arising from the \(n\)th function of the Fourier series with trigonometric function \(\phi\). When the Fourier coefficients can be calculated in closed form symbolically, we will do so. Some Fourier coefficients have to be calculated through numerical integration. The function condI represents the inhomogeneous part of the linear equation arising from the incoming wave.

We proceed in a completely analogous manner for the right arc (where there is no inhomogeneous part).

\begin{verbatim}
In[9]:= Block[{\phi = 0.4},
       Show[GraphicsArray[Block[{{DisplayFunction = Identity},
                                  Plot[Evaluate[Table[tLeft[\#, n, \phi], \{n, 1, 7, 2\}],
                                   \{\phi, Pi - \phi, Pi + \phi\}, Frame -> True, Axes -> False] &/@\{Cos, Sin\}]]]]
\end{verbatim}

\begin{verbatim}
In[10]:= Do[cs = \{Cos, Sin\}[\{\beta\}];
       (* center disk -- symbolically calculatable *)
       cond\[left, \alpha, \mu\_?NonNegative, \{cs, n\_\}] = Integrate[
         (D[u[\mu, \{r, \phi\}], \{r, \alpha\}] /. r \to R) tLeft[cs, n, \phi], \{\phi, Pi - \phi, Pi\}];
       (* center disk \(\mu = 0\) *)
       cond\[left, \alpha, 0, \{cs, 0\}] =
         Limit[cond\[left, \alpha, 0, \{cs, n\}], n \to 0];
       (* left arm -- numerically calculatable *)
       cond\[left, \alpha, \mu\_?(-1 \geq \# \geq -p) \&, \{cs, n\_\}] := NIntegrate[
         Evaluate[D[-\chi\L[-m, \{r Cos[\phi], r Sin[\phi]\}], \{r, \alpha\}] /. r \to R]
         tLeft[cs, n, \phi], \{\phi, Pi - \phi, Pi\}];
       (* right arm coefficients are absent *)
       cond\[left, \alpha, \mu\_?(-p - 1 \geq \# \geq -p - q) \&, \{cs, n\_\}] := 0;
       (* inhomogeneous part -- numerically calculatable *)
       condI\[Left, \alpha, \{cs, n\_\}] := NIntegrate[Evaluate[
         (D[\chi R[i, \{r Cos[\phi], r Sin[\phi]\}], \{r, \alpha\}] /. r \to R) tLeft[cs, n, \phi],
         \{\phi, Pi - \phi, Pi\}], \{\alpha, 0, 1\}, \{\beta, 1, 2\}]
\end{verbatim}
### Matrix Construction

We have to determine \( w = p + (o + 1) + q \) coefficients \( c_k \). To find them, we will have to use more equations than variables. (If we take the same number of equations as unknown coefficients, we would get a unique solution that makes the specified Fourier coefficients vanish but which could oscillate dramatically. See the following explicit numerical results.) Here, we will use \( d \) Fourier modes along the circular arcs and \( e \) modes along the longer Dirichlet parts of the center disk.

As mentioned earlier, we will use the cos-modes along the arcs joining the center disk with the left and right arm and the sin-modes along the Dirichlet part of the center disk. We could, of course, also use some of the sin modes along the joining arcs.

The function `makeLHSAndRHS[prec, \{d, e\}]` will construct the coefficient matrix and the right-hand side of the system of linear equations for the \( c_k \) with \( 4d + e \) equations.
Function and Gradient Values

A solution for the coefficients \( c_k \) gives us access to the solution \( \psi(x, y) \) in the three regions. Using the transformation rules for changing from derivatives in Cartesian to derivatives in polar coordinates

\[
\frac{\partial}{\partial x} = \frac{\partial \varphi}{\partial x} + \frac{\partial}{\partial r} \left( \frac{\varphi}{r} \right) - \frac{\sin(\varphi)}{r} \frac{\partial \varphi}{\partial r} + \cos(\varphi) \frac{\partial}{\partial r}
\]

\[
\frac{\partial}{\partial y} = \frac{\partial \varphi}{\partial y} + \frac{\partial}{\partial r} \left( \frac{\varphi}{r} \right) + \frac{\cos(\varphi)}{r} \frac{\partial \varphi}{\partial r} + \sin(\varphi) \frac{\partial}{\partial r},
\]

we can implement a function, makeWaveFunctionAndGradientDefinition, that sets up definitions for the solution \( \Psi = \psi(x, y) \) and the gradient vector \( \nabla \Psi = (\partial_x \psi(x, y), \partial_y \psi(x, y)) \). (We will use the gradient vector to calculate currents associated with the complex-values wave function \( \psi(x, y) \).)
makeWaveFunctionAndGradientDefinition[
  {ψ, nablaψ}, sol, prec, opts___] :=
  Module[
    {as, ys, βs, ψLeft, ψRight, besselValues, cosValues, dataDot, dataDxIpoDot, ψDot, nablaψLeft, nablaψRight, besselValues, cosValues, dataDxDot, dataDyDot, dataDxIpoDot, dataDyIpoDot, nablaψDot, ppDot = 120, ε = 10^-6},
    (* extract coefficients for left arm, right arm, and center disk *)
    as = Reverse@Take[sol, {q + 1, q + p}];
    ys = Reverse@Take[sol, q];
    βs = Take[sol, -ω - 1];
    （* form solutions in left and right arms *)
    ψLeft[x_, y_] := χR[1, {x, y}] + Table[χ[m, {x, y}], {m, p}];
    ψRight[x_, y_] := Table[χR[m, {x, y}], {m, q}].ys;
    （* form solution in center disk; minimize number of relatively time-consuming Bessel function evaluations *)
    besselValues = Table[
      {r, Table[N[BesselJ[μ, Sqrt[ε] r], prec], {μ, 0, Length[βs] - 1}]},
      {r, ε, R, (R - ε)/ppDot}];
    cosValues = Table[
      {φ, Table[N[Cos[μ φ], prec], {μ, 0, Length[βs] - 1}]},
      {φ, 0, Pi, Pi/ppDot}];
    dataDot = Table[
      besselValues[1, 1], cosValues[1, 1],
      besselValues[1, 2], cosValues[2, 1]], {i, Length[besselValues]}, {j, Length[cosValues]} // Flatten[#, 1] &;
    （* interpolate values *)
    dataIpoDot = Interpolation[
      dataDot, InterpolationOrder -> 5];
    （* Cartesian coordinate system values inside center disk *)
    ψDot[x_, y_] :=
      dataIpoDot[Sqrt[x^2 + y^2], If[x == y == 0, 0, ArcTan[x, y]]];
    （* globally visible solution *)
    ψ[x_, y_] :=
      Which[Re[BesselJ[μ - 1, Sqrt[ε] r]] ηBesselJ[μ - 1, Sqrt[ε]],
        Re[N[x^2 + y^2] - ρ R^2, ψDot[x, y],
          Re[N[x]] ≤ N[1 - R Cos[φ]]],
          Left[ψ[x, y], Re[N[x]] ≥ N[1 - R Cos[φ]],
          ψRight[x, y], True, ψDot[x, y]];
    If[Not[TrueQ[MakeGradientDefinition /. {opts}]],
      （* form solutions gradients in left and right arms *)
      nablaψLeft[x_, y_] := χRDx[1, {x, y}] + Table[χLDx[m, {x, y}], {m, p}].as,
        χRDy[1, {x, y}] + Table[χLDy[m, {x, y}], {m, p}].as;
      nablaψRight[x_, y_] := Table[χRDx[m, {x, y}], {m, q}].ys,
        Table[χRDy[m, {x, y}], {m, q}].ys;
      （* form solution in center disk; minimize number of relatively time-consuming Bessel function evaluations *)
      besselValues = Table[
        {r, Table[N[Sqrt[ε]/2 BesselJ[μ - 1, r Sqrt[ε]] - BesselJ[μ + 1, r Sqrt[ε]]], prec],
          {μ, 0, Length[βs] - 1}]},
        {r, ε, R, (R - ε)/ppDot}];
      cosValues = Table[
        {φ, Table[N[-μ Sin[μ φ]], prec],
          {μ, 0, Length[βs] - 1}]},
        {φ, 0, Pi, Pi/ppDot}];
      （* make list of data of the form {r, φ, ψ[r, φ]} *)
      dataDxDot = Table[
        besselValues[1, 1], cosValues[1, 1],
        besselValues[1, 2], cosValues[2, 1]], Table[
        besselValues[1, 2], cosValues[2, 1]], Table[
        besselValues[1, 2], cosValues[2, 1]], Table[
        besselValues[1, 2], cosValues[2, 1]], Table[
        besselValues[1, 2], cosValues[2, 1]], Table[
        besselValues[1, 2], cosValues[2, 1]]]
Measuring the Quality of the Solution

Before visualizing the eigenfunctions, we should determine values of the parameters $p, q, o, d, e$, and working precisions appropriate for the energy values under consideration (say $E \approx 10 \ldots 5$).

\begin{verbatim}
(* set global parameter values *)
E = 3; R = 10; L = 5;
Clear[p, q, o, b, e, prec];
\end{verbatim}

The function `solutionQuality` carries out numerical integrations along the domain boundaries to quantify the quality of the solutions.
In[24]:= solutionQuality[{{1, nabla1_}}] :=
With[{
ε = 10^-6, 
 opts = Sequence[PrecisionGoal -> 3],
(* deviations from Dirichlet boundary conditions *)
 NIntegrate[Abs[ε1[R Cos[θ], R Sin[θ]]]^2, {θ, φ, Pi - θ}, opts] +
(* deviations from continuity along arcs connecting center disk and arms *)
 NIntegrate[Abs[ε1[[R - ε) Cos[θ], (R - ε) Sin[θ]] -
 ε1[(R + ε) Cos[θ], (R + ε) Sin[θ]]]^2, {θ, 0, φ}, opts] +
 NIntegrate[Abs[ε1[[R - ε) Cos[θ], (R - ε) Sin[θ]] -
 ε1[(R + ε) Cos[θ], (R + ε) Sin[θ]]]^2, {θ, Pi - φ, Pi}, opts] +
(* deviations from differentiability along arcs connecting center disk and arms *)
 NIntegrate[Abs[*use directional derivatives *]
 (nablaε1[[R - ε) Cos[θ], (R - ε) Sin[θ]] - nablaε1[[R + ε) Cos[θ],
 (R + ε) Sin[θ]]].{Cos[θ], Sin[θ]}]^2, {θ, 0, φ}, opts] +
 NIntegrate[Abs[(nablaε1[[R - ε) Cos[θ], (R - ε) Sin[θ]] -
 nablaε1[[R + ε) Cos[θ], (R + ε) Sin[θ]]].{Cos[θ], Sin[θ]}]^2,
 {θ, Pi - φ, Pi}, opts]) / (* boundary length *)
(R (2 Pi + 4 φ))

The function multiParameterSolutionQuality constructs and solves the matrices and then measures the quality of the solutions for a set of parameters. The actual solution of the linear system of overdetermined equations is achieved using the function PseudoInverse. The function multiParameterSolution Quality returns lists of the form:

{parameterValue, {linearEquationResidue, integratedBoundaryValuesDifferences,
 reflectionAndTransmissionCoefficientMinius1}}

The value integratedBoundaryValuesDifferences is the result of applying solutionQuality. Quality to the solution and reflectionAndTransmissionCoefficientMinius1 is \(||R||^2 + |T|^2 - 1\). The last quantity is expected to vanish for the exact solution.
We now carry out some experiments to determine appropriate parameters. We start by determining the number of Fourier modes versus the number of basis states. The following inputs show that 5 to 10 more Fourier modes give good results.

\(\text{In}[25]:=\) multiParameterSolutionQuality[{{pp, qq, oo, prec}, {dd, ee}, {iVar, iVar1, iVar2, iVarStep}, opts___}] :=
Module[{lhs, rhs, psi, sol}, Table[
(* set parameter values *){p, o, q} = {pp, oo, qq}; {d, e} = {dd, ee};
(* construct matrices *){lhs, rhs} = makeLHSAndRHS[prec, {d, e}];
(* solve for expansion coefficients *)psi = PseudoInverse[lhs];
(* analyze vector solution *){iVar, iVar1, iVar2, iVarStep} = make3DPlot /. {opts},
If[MatrixQ[psi], (* form solution *)sol = psi.rhs;
(* form solution and gradient *)makeWaveFunctionAndGradientDefinition[
{d, e}, nabla1, sol, prec];
(* potentially show scaled solution *)
If[MatrixQ[psi], (* form solution *)sol = psi.rhs;
(* measure solution quality *)
Off[NIntegrate::ploss];
N[(* quality of solution of the linear system of equations *)
Norm[lhs.sol - rhs]/Length[rhs],
(* quality of boundary condition fulfillment of the eigenfunction *)solutionQuality[
{d, e}, nabla1],
(* sum of absolute value of reflection and transmission coefficient *)Abs[sol[[q]]^2 + Abs[sol[[p + q]]]^2], 3, 
{1}, {iVar, iVar1, iVar2, iVarStep}]]]

We continue by determining the number of basis states needed in the arms and in the center disk. Obviously, the more basis states we take into account, the better the resulting solution. But after considering four states in the arms and about 30 in the center disk, the solution gets better only slowly. For mainly graphical purposes, we obtain a satisfactory quantitative solution.
The Mathematica Journal 10 | © 2006 Wolfram Media, Inc.
Example Solution

For some further visualizations, we will use the following solution.

```mathematica
(* geometry parameters *)
R = 10; L = 5; E = 5/2;

(* appropriate parameter values for the numerical solution *)
{p, o, q} = {5, 60, 5};
{d, e} = {5, 60};
prec = 80;

{lhs, rhs} = makeLHSAndRHS[prec, {d, e}];
```

This visualizes the left-hand side matrix and the right-hand side vector.

```mathematica
Show[GraphicsArray[Block[{DisplayFunction = Identity},
(* matrix and right-hand side vector *)
{ArrayPlot[Abs[lhs] // Transpose, PlotRange -> All],
ListPlot[Abs[rhs], Frame -> True, Axes -> False]]]]
```

```mathematica
psi = PseudoInverse[lhs];
sol = If[MatrixQ[psi], psi.rhs, $Failed];
```

The left graphic is the plot of the solution vector. It clearly shows the two initial sequences belonging to the expansion coefficients in the left and right arms. While the expansion coefficients in the center disk are increasing in absolute value, their influence is decreasing because the factor $J_m(\sqrt{E} \ r)$ decreases quickly with $m$ for $r \approx R$. From the absolute size of the coefficients, we also see that the use of high-precision numbers with $\approx 30$ digits is appropriate.

The right graphic shows the product of the expansion coefficients with the ansatz functions $J_m(\sqrt{E} \ r)$. We see that the influence of the largest modes in the function value at the center disk boundary is approximately $10^{-3}$. 
polygons in the center disk

Next, we want to visualize the solution \( \psi(x, y) \). The following auxiliary functions construct polygons adapted to the shape of the regions for a given solution \( \Psi \). Just like the joining curves, we use the arcs of the boundary of the center disk that touch the left and right arms.

```
makePolygons[points_] := (* make polygons from array of points *)
  Table[Polygon[{points[[i, j]],
    points[[i + 1, j]], points[[i + 1, j + 1]], points[[i, j + 1]]}],
  {i, Length[points] - 1}, {j, Length[points[[1]]] - 1}];
leftPolygons[\( \Psi \), \( F \)] := (* polygons in the left arm *)
makePolygons[Table[{x, y, F[\( \Psi \)[x, y]]}, {y, 0, L/2, L/2/ppArm},
  {x, -2 R, -R Cos[ArcSin[y/R]], R (2 - Cos[ArcSin[y/R]])/ppArm}]]
rightPolygons[\( \Psi \), \( F \)] := (* polygons in the right arm *)
makePolygons[Table[{x, y, F[\( \Psi \)[x, y]]}, {y, 0, L/2, L/2/ppArm},
  {x, R Cos[ArcSin[y/R]], 2 R, R (2 - Cos[ArcSin[y/R]])/ppArm}]]
dotPolygons[\( \Psi \), \( F \)] := (* polygons in the center disk *)
makePolygons[Table[{x Cos[\( \Psi \)], x Sin[\( \Psi \)], F[\( \Psi \)[x Cos[\( \Psi \)], x Sin[\( \Psi \)]]},
  {x, 0, R, R/ppDot}], \( \Psi \), 0, Pi, Pi/ppDot}]]
```

The function \texttt{make3DPlot} generates a 3D plot of \( \text{reim}(\psi(x, y)) \). It makes use of the symmetry about the \( x \)-axis.

We define the corresponding wave functions and their gradients.

```
makeWaveFunctionAndGradientDefinition[\( \Psi \), nabla\( \Psi \), \( \Psi \), prec]
```

### 3D Plots of the Scattering States

Next, we want to visualize the solution \( \psi(x, y) \). The following auxiliary functions construct polygons adapted to the shape of the regions for a given solution \( \Psi \). Just like the joining curves, we use the arcs of the boundary of the center disk that touch the left and right arms.
Contour Plots

To visualize the nodal lines \(\text{Re}(\psi(x, y)) = 0\) and \(\text{Im}(\psi(x, y)) = 0\) [3], we use a contour plot. Because the wave function can be defined outside the quantum dot and the arms through interpolation, we can use the built-in function \texttt{ContourPlot}. Plot on a rectangular region containing \(\Omega\) and then cover the parts not belonging to \(\Omega\) with white polygons. The function \texttt{makeReImAbsContourPlot} shows the curves \(\text{Re}(\psi(x, y)) = 0\) in red and the curves \(\text{Im}(\psi(x, y)) = 0\) in blue. The underlying gray level contour plot shows the probability density.
Current Plots

Because we are considering a scattering problem, not only the quantum mechanical density but also the quantum mechanical current is of interest. Because the function makeReImAbsContourPlot sets up definitions for the density and
gradient, we can define the quantum mechanical current \( j \) easily through the following formula.

\[
In[51] := \quad j[x_, y_] := \text{Re}[	ext{Conjugate}[\psi1[x, y]] \nabla \psi1[x, y] / l]
\]

Here is a 3D plot of the norm of the current density for our example solution. We reuse the function make3DPlot to conveniently adapt to the shape of the region \( \Omega \).

\[
In[52] := \quad \text{make3DPlot}[\text{Identity}, \text{Function}\{\{x, y\}, \text{Norm}[j[x, \text{Sqrt}[y^2]]]/3\}]
\]

We continue with a simple vector field visualization of the current.

\[
In[53] := \quad (* \text{scaled line segment representing current direction and magnitude} \ast *)
\quad jVector[{x_, y_}] :=
\quad \text{Line}[\{(x, y), (x, y) + 0.5 \text{ArcTan}[j[x, y]] / (\pi / 2)\}]
\quad \text{We see the characteristic vortices appearing in such structures [7].}
\]

\[
In[54] := \quad \text{Off}[\text{ArcTan}::\text{indet}]; \text{Off}[\text{Graphics}::\text{gptn}];
\quad (* \text{name for later use} \ast *) jGraphic =
\quad \text{Show}[\text{Graphics}[\{#, \#/. \text{Line}[\_] \rightarrow \text{Line}[\{1, -1\} \# &/\@ l] \& @
\quad \text{Table[If[insideQ[x, y], jVector[{x, y}], \}],}
\quad \{x, -2R, 2R, 4R/160\}, \{y, 0, R, R/40\}],
\quad \text{Frame} \rightarrow \text{False}, \text{AspectRatio} \rightarrow \text{Automatic}]
\]

### Make Streamlines

Calculation of streamlines is a natural operation to carry out on a vector field. We start from the left arm.

\[
In[55] := \quad (* \text{current direction} \ast *) jN[{x_?\text{NumberQ}, y_?\text{NumberQ}}] :=
\quad #/\text{Sqrt}[#.#] @ j[x, \text{Sqrt}[y^2]]
\]
In[56]:= Off[NDSolve::"mxst"];
With[{T = 80, pps = 16},
  Show[Flatten[{{* above direction field of the current *}
    jGraphic, 
    (* streamlines in the upper-half part * )Table[
      (* starting positions; use larger density at symmetry axis *)
      X0 = \{-2 R, y^2.3/4 L/2\};
      (* solve differential equations *)n ds =
        NDSolve[{X'[t] := jN[X[t]], X[0] \[Equal] X0, X, \{t, -T, T\},
          PrecisionGoal \[Rule] 8, AccuracyGoal \[Rule] 8, MaxSteps \[Rule] 10^5}];
      (* plot streamlines *)ParametricPlot[Evaluate[X[t] /. nds[[1]]],
        Evaluate[Flatten[\{t, nds[[1, 1, 2, 1]]\}]], Frame \[Rule] True, Axes \[Rule]
        False, AspectRatio \[Rule] Automatic, DisplayFunction \[Rule] Identity,
        PlotStyle \[Rule] \{\{Hue[0.74],\}\}, PlotRange \[Rule] \{-2 R, 2 R\}, \{0, R\},
        PlotPoints \[Rule] 200, \{y, 0, 1, 1/pps\}],
        PlotRange \[Rule] \{-2 R, 2 R\}, \{-R, R\},
        DisplayFunction \[Rule]
        $DisplayFunction]]

■ Energy-Dependent Transmission

As a function of the incoming energy, the absolute value of the transmission coefficient \( T \) is of special interest. When the widths \( L \) of the left and right arms are small compared with the diameter of the center disk \( R \), a resonance is expected if the energy of the incoming wave is near an eigenvalue of the center disk with Dirichlet boundary conditions. Here we will consider the energy range \( 1 \leq E \leq 2 \). We start by calculating the lowest eigenstates of the center disk with Dirichlet boundary conditions. The exact eigenvalues for a disk with homogeneous Dirichlet boundary conditions are \( E_{jk} = \mu_{jk}/R^2 \), where \( \mu_{jk} \) is the \( k \)th zero of the Bessel function \( f_j(z) \).

In[58]:= Needs["NumericalMath'\Bessel\Zeros'\""]
The Mathematica Journal 10 | © 2008 Wolfram Media, Inc.

The Mathematica Journal 10 | © 2008 Wolfram Media, Inc.

Scattering States in a 2D Quantum Dot Embedded in a Waveguide

(* eigenvalues of the center dot with Dirichlet boundary conditions *)
dotEigenvalues = Select[Flatten[
    Table[MapIndexed[{v, #2[[1]]}, #] &,
      BesselJZeros[v, 30] ^2 / R ^2,
      {v, 0, 30}, 1, (1 ≤ # &)[-1] ≤ 2) &]
]

Out[59]=

{{0, 4}, 1.3904}, {{1, 3}, 1.03499},
{{1, 4}, 1.77521}, {{2, 3}, 1.35021}, {{3, 3}, 1.69395},
{{4, 2}, 1.22428}, {{5, 2}, 1.52241}, {{6, 2}, 1.84669},
{{7, 1}, 1.22908}, {{8, 1}, 1.49453}, {{9, 1}, 1.78337}

The function norm calculates the norm of these states.

In[60]:=

norm[{v_, _}, E_] := Sqrt[If[v == 0, 2 Pi, Pi]
   (-R ^2 / BesselJ[v - 1, R Sqrt[E]] BesselJ[v + 1, R Sqrt[E]]]]

We use the function overlap to calculate the overlap integral between the
eigenstates of the center disk and the scattering states within it. We form the
overlap integral with the expansion functions symbolically. Due to the orthogonality
of the angular parts of the eigenfunctions and the scattering state expansion
functions, we have only o + 1 terms to consider.

In[61]:=

Clear[L, R, E]

overlapFactor[E_, {v_, n_}, Eev_] =
  If[v == 0, 2 Pi, Pi] * 1 / norm[{v, n}, Eev] *
  Integrate[BesselJ[v, Sqrt[E] x] BesselJ[v, Sqrt[Eev] x] x,
    {x, 0, R}] // Simplify[#1, R > 0] &

Out[62]=

(\[Sqrt]2 \[Sqrt]Eev BesselJ[-1 + v, R Sqrt[Eev]] BesselJ[v, R Sqrt[E]] -
  \[Sqrt]E BesselJ[-1 + v, R Sqrt[E]] BesselJ[v, R Sqrt[Eev]])
  If[v == 0, 2 \[Pi], \[Pi]] / ((E - Eev)
  \[Sqrt]-BesselJ[-1 + v, R Sqrt[Eev]] BesselJ[1 + v, R Sqrt[Eev]] If[v == 0, 2 \[Pi], \[Pi]]

In[63]:=

overlap[sol_, E_, {v_, n_}, Eev_] :=
  sol[[p + q + 1 + v]] overlapFactor[E, {v, n}, Eev]

Now, we calculate the transmission data. To quickly get a rough idea of its
energy dependence, we use a smaller number of basis states.

In[64]:=

R = 10; L = 5;

prec = 40;

(* use smaller parameter values for a qualitative dependence on E *)
{p, o, q} = {3, 30, 3}; {d, e} = {4, 32};

transmissionData = Table[
    (* construct matrices *)
    lhs, rhs = makeLHSAndRHS[prec, {d, e}];
    (* construct solution vector *) psi = PseudoInverse[lhs];
    sol = If[MatrixQ[psi], psi.rhs, $Failed];
    (* extract transmission
    coefficient and form overlap matrix elements *) {{E, T = sol[[q]],
    Table[overlap[sol, E, dotEigenvalues[[k]]],
      {k, Length[dotEigenvalues]}]}, {E, 1, 2, 1/100}};
Here are the resulting transmission data and overlap integrals. The left graphic shows the transmission with the scaled absolute values of the overlap integrals. The right graphic shows the transmission with the argument of the transmission coefficient and the arguments of the overlap integrals. We clearly see a strong correlation between the transmission coefficient of the incoming wave in resonance with the eigenstates of the center disk with Dirichlet boundary conditions.

\[
\text{In[65]} = \text{Show[GraphicsArray[}
\quad \text{Function[\{f1, f2, pr\}, Show[Graphics[\{(* transmission data *)}
\quad \quad \text{Thickness[0.01], Line[\{\#[[1]], f1[\#[[2]]]\}] \& @
\quad \quad \text{transmissionData\}], \{PointSize[0.004],}
\quad \quad \text{Table[Point[\{\#\}], \{h, \{-Pi, Pi, 2 Pi / 200\}\}] \& @
\quad \quad \text{dotEigenvalues\}], \{(* overlap integrals *)}
\quad \quad \text{MapIndexed[}
\quad \quad \quad \text{Hue[0.8 \#2[[1]] / Length[dotEigenvalues]], Thickness[0.004],}
\quad \quad \quad \text{\#} \& \text{, MapIndexed[}
\quad \quad \quad \quad \text{Hue[0.8 \#2[[1]] / Length[dotEigenvalues]], Thickness[0.002],}
\quad \quad \quad \quad \text{\#} \& \text{, Line[Transpose[\{First/@ transmissionData, f2[\#]\}]] \& @}
\quad \quad \quad \quad \text{Transpose[Last/@ transmissionData]]\}], Frame -> True,}
\quad \quad \text{PlotRange -> pr, DisplayFunction -> Identity]]] \& @
\quad \quad \text{(* function to be displayed *)}
\quad \quad \{\{Abs,}
\quad \quad \quad \{3 / 2 Abs[\#] / Max[Abs[Last/@ transmissionData]] \&,}
\quad \quad \quad \{0, 3/2\},
\quad \quad \quad \{1/Pi Arg[\#] \&, 1/Pi Arg[\#] \&, \{-1, 1\}\}]]\]
\]

\[\]

### Energy-Dependent Eigenfunctions

We end our excursion with an animation showing the dependence of the eigenfunctions on the energy of the incoming wave. We use fewer expansion states than in our example to speed up the calculation. But we take enough states to get a sufficient resolution for an animation.

\[
\text{In[65]} = \{p, o, q\} = \{4, 40, 4\};
\quad \text{R = 10; L = 5;}
\quad \text{prec = 50;}
\]
animationGraphicsPair[\_] := Module[{lhs, rhs, psi},
(* construct matrices *){lhs, rhs} = makeLHSAndRHS[prec, {4, 40}];
(* construct solution vector *) psi = PseudoInverse[lhs];
sol = If[MatrixQ[psi], psi.rhs, $Failed];
(* construct wave function and gradient *)
makeWaveFunctionAndGradientDefinition[{\$1, nabla\$1}, sol, prec];
(* show 3D plot of density and contour plot of nodal lines *)
Show[GraphicsArray[{make3DPlot[ArcTan[Abs[#]^2/12]/(\[Pi]/2) &,
\$1, ViewPoint -> {0, -3, 3}, PlotLabel -> N[\$\$]},
{makeReImAbsContourPlot[\$1]}]}
]

Do[animationGraphicsPair[\$_], {\$1, 1, 3, 2/100}]

This animation shows three frames. For the full result see Corner10-1-full.nb.

**Summary**

In this Corner, we calculated eigenfunctions belonging to the continuous spectrum of an infinite 2D domain with Dirichlet boundary conditions. Using the mode-matching technique and expansion in eigenfunctions of rectangular and circular domains, we reduced the problem to the numerical evaluation of a few integrals and the calculation of pseudoinverses. Using the expansion coefficients, we calculated the eigenfunctions and the corresponding current densities. The interested reader can continue with changing the ratio of the arm width to the center disk diameter \(L/R\) or calculating Bohm trajectories of microparticles instead of the streamlines of the current vector field. Or one could search for localized normalizable eigenfunctions below the beginning of the continuum spectrum \((\pi/L)^2\) using singular value decompositions of the left-hand side matrices (no right-hand side vectors) for states symmetric or antisymmetric with respect to the y-axis. Or one could investigate the statistics of the wave function and current values within the center disk [7]. Or one could cut a hole in the center disk and investigate the influence of a Aharanov–Bohm-type magnetic field through this hole [6]. Or…
References


Additional Material

Corner10-1-full.nb

Available at www.mathematica-journal.com/issue/v10i1/download.