

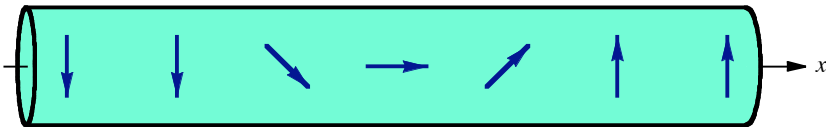
# *A Mode-Matching Method for Multichannel Scattering Problems*

**Peter Falloon**

We present a simple method for solving scattering problems with multiple incoming and outgoing channels and demonstrate its application to the problem of conduction through a ferromagnetic domain wall.

## ■ Introduction

The motivation for the work presented in this article is a problem from condensed matter physics concerning electron transport through ferromagnetic domain walls. Domain walls are spatially extended boundaries which separate magnetically homogeneous domains existing in ferromagnetic materials. They are regions in which the magnetization vector reverses by  $180^\circ$  over a length  $\lambda$  typically on the order of 10–100nm (Figure 1) [1]. Electrons travelling through such a structure experience a scattering due to the interaction of their intrinsic spin magnetic moment with the rotating magnetization. Recently, this scattering effect has been the subject of a large amount of experimental and theoretical research [2]. Much of this interest has been stimulated by the exciting potential for technological applications in solid state information storage devices that may be offered by storing domain walls in nanowire structures.



**Figure 1.** The rotating magnetization vector in a domain wall in a nanowire.

We have been involved in theoretical research aimed at understanding the physics of electron transport through domain walls in nanowires [3, 4]. Here we consider the problem of electron scattering from a domain wall using a simple

free-electron model in which the domain wall is represented as a spin-dependent potential. Taking the electron spin into account requires that we work with two-component spinor wavefunctions. This problem is analogous to one-dimensional potential scattering problems from elementary quantum mechanics, with the complication that the domain wall causes mixing between the two spinor components. There are thus multiple transmitted and reflected states.

Although a variety of computational methods exist for such problems (for example, the recursive Green's function (RGF) algorithm), it is desirable to have an intuitive and simple method. We develop such a method in this article by extending the approach commonly used for the elementary one-dimensional problem.

## □ Preliminary Code

We load the Graphics ` package and set formatting commands.

```
In[1]:= Needs["Graphics`"]
        $FormatType = TraditionalForm;
        $TextStyle =
          {TraditionalForm, FontSize -> 6, FontFamily -> "Helvetica-Oblique"};
```

For the symbolic manipulations we perform in this article it is useful to modify the behaviour of the Equal function so that functions with the Listable attribute will be distributed across expressions with head Equal.

```
In[4]:= Unprotect[Equal];
        Equal /: lhs : ((f_Symbol)?(MemberQ[Attributes[#], Listable] &))[
          ___, _Equal, ___] := Thread[Unevaluated[lhs], Equal]
        Protect[Equal];
```

We will also make use of the following two functions: ThreadEqual uses the function Thread to distribute the Equal function over nested lists to yield a one-dimensional list of equalities.

```
In[7]:= ThreadEqual[expr_Equal] :=
        Flatten@Thread[Flatten/@expr] /. {a_} == {b_} -> a == b
        ThreadEqual[expr_List] := Flatten[ThreadEqual/@expr]
```

PlotWaveFn generates a pair of graphs showing each component of a two-component wavefunction.

```
In[9]:= PlotWaveFn[Ψ_, λ_, range_List] :=
        DisplayTogetherArray[Plot[Ψ[[1, 1]], range,
          PlotStyle -> RGBColor[1, 0, 0], DisplayFunction -> Identity,
          PlotLabel -> StyleForm["Spin ↑", "SR", FontSize -> 8],
          AxesLabel -> {StyleForm["x", "S0", FontSize -> 8], ""},
          Epilog -> {Dashing[{0.01}], Line[{{-λ/2, 0}, {-λ/2, 10}}],
            Line[{{λ/2, 0}, {λ/2, 10}}]}, Plot[Ψ[[2, 1]], range,
          PlotStyle -> RGBColor[0, 0, 1], DisplayFunction -> Identity,
          PlotLabel -> StyleForm["Spin ↓", "SR", FontSize -> 8],
          AxesLabel -> {StyleForm["x", "S0", FontSize -> 8], ""},
          Epilog -> {Dashing[{0.01}], Line[{{-λ/2, 0}, {-λ/2, 10}}],
            Line[{{λ/2, 0}, {λ/2, 10}}]}]]]
```

## ■ The Mode-Matching Method

### □ Schrödinger Equation for the Domain Wall Scattering Problem

The conduction electron wavefunctions satisfy the two-component Schrödinger equation

$$-\frac{\hbar^2}{2m} \frac{\partial^2 \Psi}{\partial x^2} + \frac{\Delta}{2} \begin{pmatrix} f_z(x) & f_x(x) \\ f_x(x) & -f_z(x) \end{pmatrix} \Psi(x) = \mathcal{E} \Psi(x), \tag{1}$$

where  $\Psi = \begin{pmatrix} \psi_\uparrow \\ \psi_\downarrow \end{pmatrix}$  is the spinor wavefunction for the electron (in the  $z$  spin basis).

$\mathcal{E}$  is the total electron energy and  $\Delta$  is the energy splitting between spin  $\uparrow$  and  $\downarrow$  components. The vector  $\{f_x(x), f_y(x), f_z(x)\}$  is a unit vector parallel to the magnetization direction in the ferromagnet.

For computation, we use units with  $\frac{\hbar^2}{2m} = 1$ . We define Equation (1) as follows.

```
In[10]:= Seqn[x_] =
  (psiu''[x]) + (E (psiu[x]) + (1/2) Delta (fx[x] psiu[x] + fz[x] psiu[x]) + psiu''[x]) == 0 //
  ThreadEqual
  (psid''[x]) + (E (psid[x]) + (1/2) Delta (fx[x] psid[x] - fz[x] psid[x]) + psid''[x]) == 0 //
```

```
Out[10]:= {E psiu[x] + 1/2 Delta (fx[x] psiu[x] + fz[x] psiu[x]) + psiu''[x] == 0,
  E psid[x] + 1/2 Delta (-fz[x] psid[x] + fx[x] psid[x]) + psid''[x] == 0}
```

Outside the domain wall the magnetization is uniform with opposite direction on either side of the wall:

$$f_x(x) = 0, \quad f_z(x) = \text{sgn}(x), \quad |x| > \frac{\lambda}{2}. \tag{2}$$

We suppose the magnetization rotates in the  $xz$ -plane,  $M \propto (f_x(x), 0, f_z(x))$ . We consider two possible profiles for inside the wall:

#### 1. Trigonometric

$$f_x(x) = \cos\left(\frac{\pi x}{\lambda}\right), \quad f_z(x) = \sin\left(\frac{\pi x}{\lambda}\right), \quad |x| \leq \frac{\lambda}{2}, \tag{3}$$

```
In[11]:= fxtrig[lam_][x_] := {Cos[x/lam] - lambda/2 < x < lambda/2,
  0 Abs[x] >= lambda/2}
  fztrig[lam_][x_] := {Sin[x/lam] - lambda/2 < x < lambda/2,
  Sign[x] Abs[x] >= lambda/2}
```

#### 2. Linear

$$f_x(x) = \sqrt{1 - \left(\frac{2x}{\lambda}\right)^2}, \quad f_z(x) = \frac{2x}{\lambda}, \quad |x| \leq \frac{\lambda}{2}. \tag{4}$$

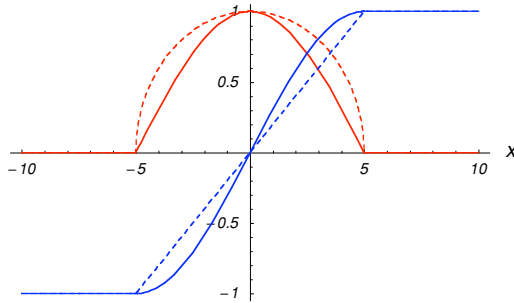
$$\text{In[13]:= } \text{fxlin}[\lambda\_][x\_]:= \begin{cases} \sqrt{1 - \left(\frac{2x}{\lambda}\right)^2} & -\frac{\lambda}{2} < x < \frac{\lambda}{2} \\ 0 & \text{Abs}[x] \geq \frac{\lambda}{2} \end{cases}$$

$$\text{fzlin}[\lambda\_][x\_]:= \begin{cases} \frac{2x}{\lambda} & -\frac{\lambda}{2} < x < \frac{\lambda}{2} \\ \text{Sign}[x] & \text{Abs}[x] \geq \frac{\lambda}{2} \end{cases}$$

The trigonometric profile is useful because it can be solved in closed-form, while the linear profile is useful in a perturbative approach since, if the  $f_x(x)$  component is ignored, solutions can be found in terms of Airy functions.

We visualize the components of the potential.

```
In[15]:= With[{λ = 10},
  Plot[{fxtrig[λ][x], fztrig[λ][x], fxlin[λ][x], fzlin[λ][x]},
    {x, -λ, λ}, PlotStyle →
    {Red, Blue, {Red, Dashing[{0.01}]}, {Blue, Dashing[{0.01}]}}],
  AxesLabel → {StyleForm["x", "S0", FontSize → 8], ""}]
```



**Figure 2.** Components of the potential. The trigonometric profiles  $f_x$  and  $f_z$  are red and blue respectively. The linear profiles  $f_x$  and  $f_z$  are red and blue dashed lines respectively.

Outside the domain wall region, solutions of Equation (1) are combinations of the following plane-wave basis states:

$$\begin{pmatrix} \exp(\pm i k_+ x) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \exp(\pm i k_- x) \end{pmatrix}, \quad x < \frac{-\lambda}{2}, \quad (5)$$

$$\begin{pmatrix} \exp(\pm i k_- x) \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \exp(\pm i k_+ x) \end{pmatrix}, \quad x > \frac{\lambda}{2}.$$

Here  $k_{\pm}$  are spin-dependent wavevectors:

$$k_{\pm} = \sqrt{\frac{2m}{\hbar^2} \left( E \pm \frac{\Delta}{2} \right)}. \quad (6)$$

□ **Scattering States and Scattering Amplitudes**

The goal of the scattering problem is to find *scattering state* solutions of Equation (1). These represent a steady-state flux of electrons incident on the scattering region which is partially transmitted and partially reflected. Specifically, we consider states incident from the left ( $x = -\infty$ ):

$$\Psi_{\uparrow}(x) = \begin{cases} \begin{pmatrix} \exp(i k_+ x) + r_{\uparrow\uparrow} \exp(-i k_+ x) \\ r_{\uparrow\downarrow} \sqrt{k_+ / k_-} \exp(-i k_- x) \end{pmatrix}, & x < \frac{-\lambda}{2} \\ \begin{pmatrix} t_{\uparrow\uparrow} \sqrt{k_+ / k_-} \exp(i k_- x) \\ t_{\uparrow\downarrow} \exp(i k_+ x) \end{pmatrix}, & x > \frac{\lambda}{2} \end{cases} \quad (7)$$

$$\Psi_{\downarrow}(x) = \begin{cases} \begin{pmatrix} r_{\downarrow\uparrow} \sqrt{k_- / k_+} \exp(-i k_+ x) \\ \exp(i k_- x) + r_{\downarrow\downarrow} \exp(-i k_- x) \end{pmatrix}, & x < \frac{-\lambda}{2} \\ \begin{pmatrix} t_{\downarrow\uparrow} \exp(i k_- x) \\ t_{\downarrow\downarrow} \sqrt{k_- / k_+} \exp(i k_+ x) \end{pmatrix}, & x > \frac{\lambda}{2}, \end{cases} \quad (8)$$

where  $r_{ab}$  and  $t_{ab}$  are the reflection and transmission amplitudes. The factors  $\sqrt{k_a / k_b}$  take account of the change in flux between regions with differing wavevectors and where  $a$  and  $b$  can each be spin up or spin down.

$$\text{In[16]:= } \Psi_{\text{upleft}}[x_] := \begin{pmatrix} \text{Exp}[i k_p x] + r_{uu} \text{Exp}[-i k_p x] \\ r_{ud} \sqrt{k_p / k_m} \text{Exp}[-i k_m x] \end{pmatrix}$$

$$\Psi_{\text{upright}}[x_] := \begin{pmatrix} t_{uu} \sqrt{k_p / k_m} \text{Exp}[i k_m x] \\ t_{ud} \text{Exp}[i k_p x] \end{pmatrix}$$

$$\Psi_{\text{dnleft}}[x_] := \begin{pmatrix} r_{du} \sqrt{k_m / k_p} \text{Exp}[-i k_p x] \\ \text{Exp}[i k_m x] + r_{dd} \text{Exp}[-i k_m x] \end{pmatrix}$$

$$\Psi_{\text{dnright}}[x_] := \begin{pmatrix} t_{du} \text{Exp}[i k_m x] \\ t_{dd} \sqrt{k_m / k_p} \text{Exp}[i k_p x] \end{pmatrix}$$

The corresponding transmission and reflection *probabilities* are defined by

$$T_{ab} = |t_{ab}|^2, \quad R_{ab} = |r_{ab}|^2. \quad (9)$$

Conservation of probability requires that these sum to unity for a given state:

$$\sum_b (T_{ab} + R_{ab}) = 1. \quad (10)$$

The functions  $\Psi_{\uparrow}(x)$  and  $\Psi_{\downarrow}(x)$  span the subspace of solutions of Equation (1) which contains all solutions with only right-moving components in the region  $x > \lambda/2$ , that is, all solutions which have no incoming component from the right.

## □ An Alternative Set of Solutions

The key idea in our method is to determine two *arbitrary* linearly independent solutions from the same space as  $\Psi_{\uparrow}(x)$  and  $\Psi_{\downarrow}(x)$  for which we choose the form in the region  $x > \lambda/2$ . For the sake of definiteness, we let

$$\Phi_1(x) = \begin{pmatrix} \exp(i k_- x) \\ \exp(i k_+ x) \end{pmatrix}, \quad \Phi_2(x) = \begin{pmatrix} \exp(i k_- x) \\ -\exp(i k_+ x) \end{pmatrix}, \quad \text{for } x > \frac{\lambda}{2}. \quad (11)$$

The choice here is arbitrary, and a more intuitive choice would perhaps be to make the transmitted functions be either up or down:

$$\Phi_1(x) = \begin{pmatrix} \exp(i k_- x) \\ 0 \end{pmatrix}, \quad \Phi_2(x) = \begin{pmatrix} 0 \\ \exp(i k_+ x) \end{pmatrix}. \quad [\text{not used}]$$

However, the choice we have made in Equation (11) ensures that both up and down components will be roughly of the same magnitude, which minimizes the possibility for numerical difficulties that might arise if one of the components were set to zero.

Because the functions  $\Phi_{1,2}(x)$  are linear combinations of  $\Psi_{\uparrow,\downarrow}(x)$ , they contain both incoming and reflected components in the region  $x \leq -\lambda/2$ , so we write

$$\Phi_{1,2}(x) = \begin{pmatrix} \alpha_{1,2} \exp(i k_+ x) + \beta_{1,2} \exp(-i k_+ x) \\ \gamma_{1,2} \exp(i k_- x) + \delta_{1,2} \exp(-i k_- x) \end{pmatrix}. \quad (12)$$

$$\begin{aligned} \text{In[20]:= } \phi1\text{left}[x\_ ] &:= \begin{pmatrix} \alpha 1 \text{Exp}[i k p x] + \beta 1 \text{Exp}[-i k p x] \\ \gamma 1 \text{Exp}[i k m x] + \delta 1 \text{Exp}[-i k m x] \end{pmatrix} \\ \phi1\text{right}[x\_ ] &:= \begin{pmatrix} \text{Exp}[i k m x] \\ \text{Exp}[i k p x] \end{pmatrix} \\ \phi2\text{left}[x\_ ] &:= \begin{pmatrix} \alpha 2 \text{Exp}[i k p x] + \beta 2 \text{Exp}[-i k p x] \\ \gamma 2 \text{Exp}[i k m x] + \delta 2 \text{Exp}[-i k m x] \end{pmatrix} \\ \phi2\text{right}[x\_ ] &:= \begin{pmatrix} \text{Exp}[i k m x] \\ -\text{Exp}[i k p x] \end{pmatrix} \end{aligned}$$

We obtain  $\Phi_{1,2}(x)$  in the interval  $-\lambda/2 < x < \lambda/2$  by solving Equation (1) numerically using the values at  $x = \lambda/2$  as initial conditions.

## □ Expressing Scattering States and Amplitudes in Terms of $\Phi_{1,2}(x)$

The scattering states  $\Psi_{\uparrow,\downarrow}(x)$  can be expressed as a linear combination of  $\Phi_{1,2}(x)$  according to

$$\Psi_{\uparrow}(x) = A_1 \Phi_1(x) + A_2 \Phi_2(x), \quad \Psi_{\downarrow}(x) = B_1 \Phi_1(x) + B_2 \Phi_2(x). \quad (13)$$

$$\begin{aligned} \text{In[24]:= } \Psi\text{eqsleft} &= \{A1 \phi1\text{left}[-\lambda/2] + A2 \phi2\text{left}[-\lambda/2] == \Psi\text{upleft}[-\lambda/2], \\ & B1 \phi1\text{left}[-\lambda/2] + B2 \phi2\text{left}[-\lambda/2] == \Psi\text{dnleft}[-\lambda/2]\}; \\ \Psi\text{eqsright} &= \{A1 \phi1\text{right}[\lambda/2] + A2 \phi2\text{right}[\lambda/2] == \Psi\text{upright}[\lambda/2], \\ & B1 \phi1\text{right}[\lambda/2] + B2 \phi2\text{right}[\lambda/2] == \Psi\text{dnright}[\lambda/2]\}; \end{aligned}$$

We obtain the coefficients  $A_{1,2}$  and  $B_{1,2}$  by equating the coefficients of the incoming wave components  $\exp(i k_{\pm} x)$  at  $x = -\lambda/2$ .

```
In[26]:= ABeqns =
  Coefficient[#, {Exp[-i kp λ / 2], Exp[-i km λ / 2]}] & /@ # & ThreadEqual //
  DeleteCases[#, True] &
```

```
Out[26]= {A1 α1 + A2 α2 == 1, A1 γ1 + A2 γ2 == 0, B1 α1 + B2 α2 == 0, B1 γ1 + B2 γ2 == 1}
```

```
In[27]:= {A1, A2, B1, B2} =
  {A1, A2, B1, B2} /. First@Solve[ABeqns, {A1, A2, B1, B2}]
```

```
Out[27]= {
   $\frac{\gamma_2}{-\alpha_2 \gamma_1 + \alpha_1 \gamma_2}$ ,  $\frac{\gamma_1}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\alpha_2}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\alpha_1}{-\alpha_2 \gamma_1 + \alpha_1 \gamma_2}$ 
}
```

The scattering amplitudes  $r_{ab}$  and  $t_{ab}$  can then be found by equating the coefficients of the appropriate scattered components at  $x = -\lambda/2$  and  $x = \lambda/2$ .

```
In[28]:= rabeqs = Coefficient[#, {Exp[i kp λ / 2], Exp[i km λ / 2]}] & /@ # & ThreadEqual //
  DeleteCases[#, True] &;
  tabeqs = Coefficient[#, {Exp[i kp λ / 2], Exp[i km λ / 2]}] & /@ # & ThreadEqual //
  DeleteCases[#, True] &;
```

In solving these equations we substitute the definitions for  $A_{1,2}$  and  $B_{1,2}$ .

```
In[30]:= {ruu, rud, rdu, rdd, tuu, tud, tdu, tdd} =
  {ruu, rud, rdu, rdd, tuu, tud, tdu, tdd} /.
  First@Solve[Join[rabeqs, tabeqs], {ruu, rud, rdu, rdd,
  tuu, tud, tdu, tdd}] // Simplify[#, {kp > 0, km > 0}] &
```

```
Out[30]= {
   $\frac{\beta_2 \gamma_1 - \beta_1 \gamma_2}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\sqrt{\frac{km}{kp}} (-\gamma_2 \delta_1 + \gamma_1 \delta_2)}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\sqrt{\frac{kp}{km}} (\alpha_2 \beta_1 - \alpha_1 \beta_2)}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\alpha_2 \delta_1 - \alpha_1 \delta_2}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,
   $\frac{\sqrt{\frac{km}{kp}} (\gamma_1 - \gamma_2)}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ ,  $\frac{\gamma_1 + \gamma_2}{-\alpha_2 \gamma_1 + \alpha_1 \gamma_2}$ ,  $\frac{\alpha_1 - \alpha_2}{-\alpha_2 \gamma_1 + \alpha_1 \gamma_2}$ ,  $\frac{\sqrt{\frac{kp}{km}} (\alpha_1 + \alpha_2)}{\alpha_2 \gamma_1 - \alpha_1 \gamma_2}$ 
}
```

### □ Solving for $\Phi_{1,2}(x)$ and $\Psi_{\uparrow,\downarrow}(x)$

Now that we have an expression for the amplitudes  $r_{ab}$  and  $t_{ab}$  in terms of  $\alpha_{1,2}$ ,  $\beta_{1,2}$ ,  $\gamma_{1,2}$ , and  $\delta_{1,2}$  all that remains is to determine these coefficients. We do this by numerically solving the Schrödinger equation, Equation (1), for  $\Phi_{1,2}(x)$ . The equation is solved as an initial value problem with the value of  $\Phi_{1,2}(\lambda/2)$  and  $\Phi'_{1,2}(\lambda/2)$  as initial conditions.

```
In[31]:= init1 = {
  {ψu[λ / 2], ψd[λ / 2]} == #1right[λ / 2],
  {ψu'[λ / 2], ψd'[λ / 2]} == #1right'[λ / 2]} // ThreadEqual;
  init2 = {
  {ψu[λ / 2], ψd[λ / 2]} == #2right[λ / 2], {ψu'[λ / 2], ψd'[λ / 2]} == #2right'[λ / 2]} //
  ThreadEqual;
```

Here we assume a trigonometric domain wall and a particular set of parameter values.

```

In[33]:= E = 1.0; Δ = 0.1; λ = 10.0;
          fx[x_] = fxtrig[λ][x]; fz[x_] = fztrig[λ][x];
          kp = √(E + Δ/2); km = √(E - Δ/2);

In[36]:= ϕ1wall[x_] = ( ψu[x]
                       ψd[x] ) /.
          First@NDSolve[Join[Seqn[x], init1], {ψu, ψd}, {x, -λ/2, λ/2}]

Out[36]= {{InterpolatingFunction[{{-5., 5.}}, <>][x]},
          {InterpolatingFunction[{{-5., 5.}}, <>][x]}}

In[37]:= ϕ2wall[x_] = ( ψu[x]
                       ψd[x] ) /.
          First@NDSolve[Join[Seqn[x], init2], {ψu, ψd}, {x, -λ/2, λ/2}]

Out[37]= {{InterpolatingFunction[{{-5., 5.}}, <>][x]},
          {InterpolatingFunction[{{-5., 5.}}, <>][x]}}

```

The function `NDSolve` returns solutions in the form of `InterpolatingFunction` objects, which can be differentiated like ordinary functions.

We next use the value of these numerical solutions and their first derivative at  $x = -\lambda/2$  to find the coefficients in the representation of  $\Phi_{1,2}(x)$  for  $x \leq -\lambda/2$ , using the fact that the wavefunctions are continuous at  $x = \pm\lambda/2$ .

```

In[38]:= Clear[α1, β1, γ1, δ1, α2, β2, γ2, δ2];
          {α1, β1, γ1, δ1, α2, β2, γ2, δ2} =
          {α1, β1, γ1, δ1, α2, β2, γ2, δ2} /. First@Solve[
          {ϕ1left[-λ/2] == ϕ1wall[-λ/2], ϕ1left'[-λ/2] == ϕ1wall'[-λ/2],
          ϕ2left[-λ/2] == ϕ2wall[-λ/2], ϕ2left'[-λ/2] == ϕ2wall'[-λ/2]},
          {α1, β1, γ1, δ1, α2, β2, γ2, δ2}]

Out[39]= {1.02266 - 0.146829 i, -0.0398946 + 0.0204621 i,
          0.95478 - 0.14655 i, 0.0419418 + 0.00757023 i,
          0.905043 + 0.148117 i, -0.0398946 - 0.000755258 i,
          -1.07184 - 0.147905 i, -0.0419418 - 0.0131478 i}

```

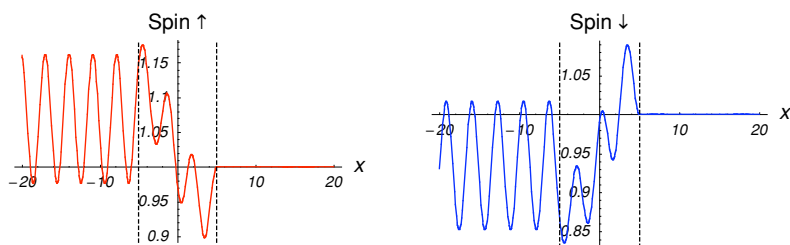
We now have the two solutions  $\Phi_{1,2}(x)$  for all  $x$ .

$$\text{In[40]:= } \phi_1[x_] := \begin{cases} \phi_{1\text{left}}[x] & x \leq -\frac{\lambda}{2} \\ \phi_{1\text{wall}}[x] & -\frac{\lambda}{2} < x < \frac{\lambda}{2} \\ \phi_{1\text{right}}[x] & \frac{\lambda}{2} \leq x \end{cases}$$

$$\phi_2[x_] := \begin{cases} \phi_{2\text{left}}[x] & x \leq -\frac{\lambda}{2} \\ \phi_{2\text{wall}}[x] & -\frac{\lambda}{2} < x < \frac{\lambda}{2} \\ \phi_{2\text{right}}[x] & \frac{\lambda}{2} \leq x \end{cases}$$

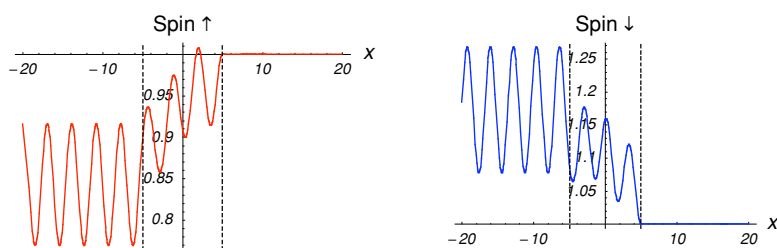
We visualize the squared magnitude of up and down components using the function `PlotWaveFn` defined at the start of this article.

```
In[42]:= PlotWaveFn[Abs[ϕ1[x]]2, λ, {x, -2 λ, 2 λ}]
```



**Figure 3.** Squared magnitude of up and down components of  $\Phi_1(x)$  are red and blue respectively.

```
In[43]:= PlotWaveFn[Abs[ϕ2[x]]2, λ, {x, -2 λ, 2 λ}]
```



**Figure 4.** Squared magnitude of up and down components of  $\Phi_2(x)$  are red and blue respectively.

Both components display oscillatory behaviour for  $x < -\lambda/2$ , since the functions contain incoming and reflected components in each spin channel.

$A_{1,2}$  and  $B_{1,2}$  now evaluate to numerical quantities,

```
In[44]:= {A1, A2, B1, B2}
```

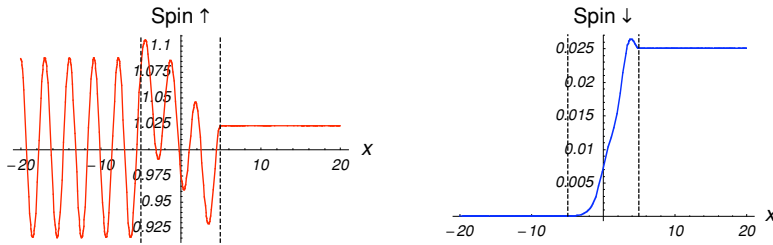
```
Out[44]= {0.535037 + 0.0731058 i, 0.476418 - 0.0737744 i,
          0.45179 + 0.0733223 i, -0.510295 + 0.0739583 i}
```

so we can immediately define  $\Psi_{\uparrow,\downarrow}(x)$  in terms of  $\Phi_{1,2}(x)$ .

```
In[45]:= Ψup[x_] = A1 ϕ1[x] + A2 ϕ2[x];
          Ψdn[x_] = B1 ϕ1[x] + B2 ϕ2[x];
```

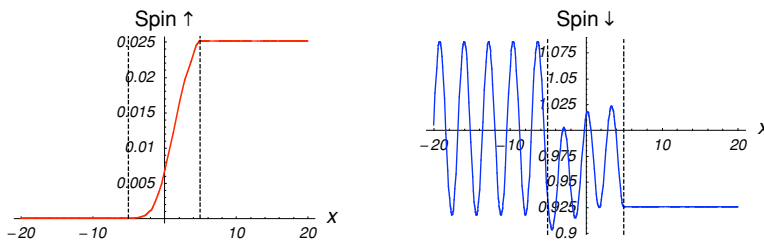
Now we plot the two spin components of each state.

```
In[47]:= PlotWaveFn[Abs[Ψup[x]]2, λ, {x, -2 λ, 2 λ}]
```



**Figure 5.** Spin components of  $\Phi_{\uparrow}(x)$  are red and blue respectively.

```
In[48]:= PlotWaveFn[Abs[Ψdn[x]]2, λ, {x, -2 λ, 2 λ}]
```



**Figure 6.** Spin components of  $\Phi_{\downarrow}(x)$  are red and blue respectively.

We also have the scattering coefficients. Here are the corresponding probabilities.

```
In[49]:= MatrixForm/@Abs[{{(tuu tdu), (ruu rdu)}, (tud tdd), (rud rdd)}]2
```

```
Out[49]= {{(0.973106 0.0251144), (0.00186855 0.0000156497)}, (0.02501 0.973106), (0.0000156496 0.00176417)}
```

A simple estimate of the error in our result is given by checking the deviation from conservation of probability (the sum of transmission and reflection should be unity).

```
In[50]:= Total/@Abs[{{ruu, rud, tuu, tud}, {rdd, rdu, tdd, tdu}}]2 - 1
```

```
Out[50]= {-2.77978 × 10-7, -3.04985 × 10-7}
```

## □ Package Implementation

The algorithm to compute the transmission and reflection coefficients of a domain wall that is described in this section has been implemented in the *Domain-Wall* package, which contains the following function.

DWCoefficients[ $E_{-}$ ,  $\Delta_{-}$ ,  $\lambda_{-}$ ]: returns the matrices of transmission and reflection probabilities for a domain wall with given parameters.

This function takes the following option.

`WallType`: specifies form of domain wall profile. Possible values are `Trig`, and `Linear`; the default is `Trig`.

Options for the `NDSolve` function (used to compute  $\Phi_{1,2}(x)$ ) can also be passed to `DWCoefficients`. This allows the accuracy and precision of the solutions to be controlled by appropriately setting the options `AccuracyGoal` (or `PrecisionGoal`) and `WorkingPrecision`.

## ■ Results

We now show some results using `DomainWall.m`. We first load the package.

```
In[51]:= AppendTo[$Path, DirectoryName[ToFileName[
          "FileName" /. NotebookInformation[EvaluationNotebook[]]]]];
In[52]:= << DomainWall`
```

## □ Transmission and Reflection Functions

The electron transport through the domain wall can be understood by studying the transmission and reflection probabilities as a function of the physical parameters. Here we briefly consider the behaviour as a function of  $\lambda$ , the domain wall width, and  $\mathcal{E}$ , the electron energy.

We work with fixed  $\Delta$  throughout.

```
In[53]:= Δ = 0.1;
```

### **Behaviour as a Function of $\lambda$**

We first fix  $\mathcal{E}$  and create a list of transmission and reflection matrices using `DWCoefficients` for a range of  $\lambda$  values.

```
In[54]:= E = 1;
          λlist = Range[1, 300, 5];
          trlist = DWCoefficients[#, E, Δ] & /@ λlist;
```

We then create suitable sublists of `trlist` by extracting the coefficients of interest (we focus on the transmission coefficients  $t_{\uparrow\uparrow}$  and  $t_{\uparrow\downarrow}$  relating to  $\Psi_+(x)$ , the incoming spin  $\uparrow$  state; results for  $\Psi_-(x)$  are similar). A useful way to extract the same part from a list of expressions is using `Part` (`[[]]`) with the first argument set to `All`.

```
In[57]:= tuulist = Transpose[{λlist, trlist[[All, 1, 1, 1]]}];
          tudlist = Transpose[{λlist, trlist[[All, 1, 2, 1]]}];
```

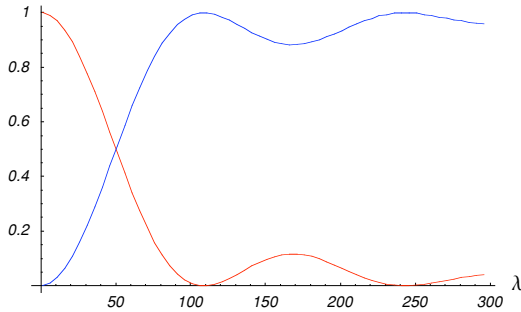
We then plot the functions.

```

In[59]:= SetOptions[MultipleListPlot, PlotJoined → True,
          PlotRange → All, PlotStyle → {Red, Blue},
          AxesLabel → {StyleForm["λ", FontSize → 8], ""}, SymbolShape → None];

MultipleListPlot[tuulist, tudlist]

```



**Figure 7.** Transmission coefficients  $t_{\uparrow\uparrow}$  and  $t_{\downarrow\downarrow}$  as a function of  $\lambda$  are red and blue respectively.

From the physics point of view, this graph demonstrates the following features:

1. Overall, the transmission with spin-reversal increases with the wall width  $\lambda$ .
2. There are complementary oscillations in  $t_{\uparrow\uparrow}$  and  $t_{\downarrow\downarrow}$  as a function of  $\lambda$ .

Both features are explained by understanding that the electron traversing the domain wall undergoes a “quasi-precessional” motion comprising precession about the local magnetization direction combined with rotation following the magnetization.

### **Behaviour as a Function of $\mathcal{E}$**

We now fix  $\lambda$  and consider the behaviour as a function of  $\mathcal{E}$ , following the same approach as in the previous case.

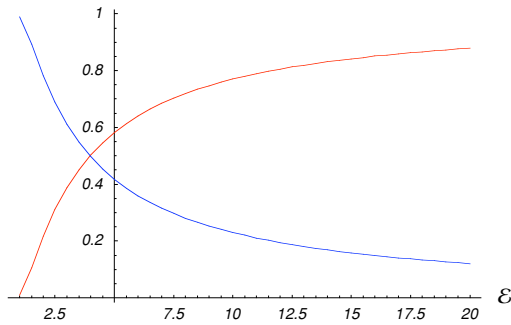
```

In[61]:= λ = 100.;
          Elist = Range[1, 20, 0.5];
          trlist2 = DWCoefficients[λ, #, Δ] & /@ Elist;
          tuulist2 = Transpose[{Elist, trlist2[[All, 1, 1, 1]]}];
          tudlist2 = Transpose[{Elist, trlist2[[All, 1, 2, 1]]}];

```

We again plot  $t_{\uparrow\uparrow}$  and  $t_{\downarrow\downarrow}$ , this time as a function of  $\mathcal{E}$ .

```
In[66]:= MultipleListPlot[tuulist2, tudlist2,
  AxesLabel -> {StyleForm["E", FontSize -> 8], ""}]
```



**Figure 8.** Transmission coefficients  $t_{\uparrow\uparrow}$  and  $t_{\uparrow\downarrow}$  as a function of  $\mathcal{E}$  are red and blue respectively.

This shows that  $t_{\uparrow\uparrow} \rightarrow 1$  for increasing  $\mathcal{E}$ . The reason for this is that for large  $\mathcal{E}$  the electron travels very rapidly through the domain wall and thus is less influenced by the rotating magnetization in the wall.

## □ Comparison of Domain Wall Profiles

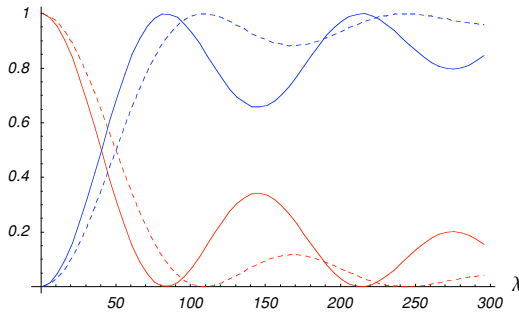
A useful application of our code is to compare the results of the trigonometric and linear models. Each model has practical advantages in analytical work, but it is important to know to what extent the results of the two differ. To check this, let us recalculate the transmission curves as a function of  $\lambda$ , this time using the linear profile.

```
In[67]:= trlist3 = DWCoefficients[#, E, \Delta, WallType -> Linear] & /@ \lambda list;
```

```
In[68]:= tuulist3 = Transpose[{\lambda list, trlist3[[All, 1, 1, 1]]}];
  tudlist3 = Transpose[{\lambda list, trlist3[[All, 1, 2, 1]]}];
```

Here we plot  $t_{\uparrow\uparrow}$  and  $t_{\uparrow\downarrow}$  for the linear wall, showing the trigonometric wall results with dashed lines.

```
In[70]:= MultipleListPlot[tuulist3, tudlist3, tuulist, tudlist, PlotStyle ->
  {Red, Blue, {Red, Dashing[{0.01}]}, {Blue, Dashing[{0.01}]}}]
```



**Figure 9.** Transmission coefficients  $t_{\uparrow\uparrow}$  and  $t_{\uparrow\downarrow}$  as a function of  $\lambda$  are red and blue respectively, with the trigonometric wall results indicated using dashed lines.

The linear wall thus shows the same qualitative features as the trigonometric wall but with quantitative differences. In particular, the oscillations are more pronounced in the linear wall, which is related to the fact that the edges of the linear wall are more abrupt than those of the trigonometric wall ( $f'_z(x)$  is discontinuous at  $x = \pm\lambda/2$  for the linear wall, but is continuous for the trigonometric wall) (Figure 2).

## ■ Conclusion

We have presented a mode-matching method which can be used for scattering problems in which there are multiple scattering channels. The method makes use of `NDSolve` to find solutions of an arbitrary scattering potential using particular arbitrary combinations of asymptotic (transmitted) components as initial values at the right-hand boundary of the scattering region. The coefficients for the corresponding incoming and reflected components of these solutions are extracted using `Solve` at the opposite boundary of the scattering region. The true scattering states are then found by taking appropriate linear combinations so as to leave only a single incoming component.

The method has the advantage of being intuitive and straightforward to implement. It permits the rapid solution of nontrivial scattering problems for which more sophisticated approaches (such as the RGF algorithm) are often called on. Furthermore, because we use `NDSolve`, results can readily be obtained to arbitrary precision with minimal effort.

The method can be directly applied to any scattering problem in which there are multiple incoming and outgoing channels. In this article we have presented the example of a spin-dependent potential in one dimension, where the channels are spin components of the electron wavefunction. However, the same approach could also be applied to, for example, a two- or three-dimensional wire where the

channels would correspond to distinct transverse wavefunction components. Thus, we expect this method to be of quite general utility.

Finally, we mention several limitations of our method. The first concerns the case where solutions in the scattering region have *evanescent* components, that is, components which grow or decay exponentially. Here the actual scattering states usually only contain a nonvanishing contribution from the decaying component. However, the initial states we calculate in general contain both decaying and growing components, which leads to a rapid loss of precision. Because of this, the existence of evanescent modes in the system tends to necessitate large working precision. This problem does not occur with more advanced approaches, such as the RGF method. Secondly, we note that since each additional channel contributes an additional dimension to be solved numerically, the difficulty of the calculation increases rapidly with the number of channels. Thus this method is not well suited to problems with a large number of channels.

## ■ References

- [1] A. Hubert and R. Schäfer, *Magnetic Domains*, New York: Springer-Verlag, 1998.
- [2] U. Ebels, A. Radulescu, Y. Henry, L. Piraux, and K. Ounadjela, "Spin Accumulation and Domain Wall Magnetoresistance in 35nm Co Wires," *Physical Review Letters*, **84**(5), 2000 pp. 983–986.
- [3] V. A. Gopar, D. Weinmann, R. A. Jalabert, and R. L. Stamps, "Electronic Transport through Domain Walls in Ferromagnetic Nanowires: Coexistence of Adiabatic and Nonadiabatic Spin Dynamics," *Physical Review B*, **69**, 2004 pp. 014426–014436. DOI-Link: [dx.doi.org/10.1103/PhysRevB.69.014426](https://doi.org/10.1103/PhysRevB.69.014426)
- [4] P. E. Falloon, R. A. Jalabert, D. Weinmann, and R. L. Stamps, "A Circuit Model for Domain Walls in Ferromagnetic Nanowires: Application to Conductance and Spin Transfer Torques," *Physical Review B*, **70**, 2004 pp. 174424–174434. DOI-Link: [dx.doi.org/10.1103/PhysRevB.70.174424](https://doi.org/10.1103/PhysRevB.70.174424)

## ■ Additional Material

DomainWall.m

Available at [www.mathematica-journal.com/issue/v10i3/download](http://www.mathematica-journal.com/issue/v10i3/download).

### About the Author

Peter Falloon completed his Ph.D. in theoretical condensed matter physics in co-tutelle between The University of Western Australia and the Université Louis Pasteur at Strasbourg in 2006. He also holds B.S. (Honours) and M.S. degrees from The University of Western Australia.

In his master's dissertation Falloon developed a *Mathematica* package to compute spheroidal wavefunctions, which will be incorporated into the next version of the

*Mathematica* system. In 2001 he completed a summer internship at Wolfram Research and from 2001 to 2004 continued to work as a consultant.

**Peter Falloon**

*School of Physics, M013*

*The University of Western Australia*

*35 Stirling Highway*

*Crawley WA 6009, Australia*

*pfalloon@gmail.com*