OPEN ACCESS

The Journal of Engineering and Exact Sciences – jCEC, Vol. 07 N. 03 (2021)
journal homepage: https://periodicos.ufv.br/ojs/jcec
ISSN: 2527-1075

Algorithms for Non-Relativistic Quantum Integral Equation

Algoritmos para Equação Integral Quântico Não Relativístico

Article Info:
Article history: Received 2021-06-21 / Accepted 2021-07-26 / Available online 2021-07-26
doi: 10.18540/jcecvl7iss3pp12699-01-19e

Jorge Henrique Sales
ORCID: https://orcid.org/0000-0003-1992-3748
Universidade Estadual de Santa Cruz, DCET/ PPGMC
Rod. Jorge Amado - 45662-900 - Ilhéus, BA, Brazil
E-mail: jhosales@uesc.br
Pedro Henrique Sales Girotto
ORCID: https://orcid.org/0000-0003-0444-9465
Centro Universitário do Estado do Pará,
Av. Alcindo Cacela n.1523, CEP: 66040 – 020, Belém, Brazil
E-mail: pedro.girotto@prof.cesupa.br

Abstract

In low energy scattering in Non-Relativistic Quantum Mechanics, the Schödinger equation in integral form is used. In quantum scattering theory the wave self-function is divided into two parts, one for the free wave associated with the particle incident to a scattering center, and the emerging wave that comes out after the particle collides with the scattering center. Assuming that the scattering center contains a position-dependent potential, the usual solution of the integral equation for the scattered wave is obtained via the Born approximation. Assuming that the scattering center contains a position-dependent potential, the usual solution of the integral equation for the scattered wave is obtained via the Born approximation. The methods used here are arbitrary kernels and the Neumann-Born series. The result, with the help of computational codes, shows that both techniques are good compared to the traditional method. The advantage is that they are finite solutions, which does not require Podolsky-type regularization.

Keywords: Quantum scattering. Fredholm. Neumann-Born. Computational modeling.

Resumo

No espalhamento a baixa energia na Mecânica Quântica não Relativística, usa-se a equação de Schödinger na forma integral. Na teoria do espalhamento quântico a autofunção de onda é dividida em duas partes, uma para a onda livre associada a partícula incidente à um centro espalhador, e a onda emergente que sai depois da partícula colidir com o centro espalhador. Admitindo que o centro espalhador contém um potencial dependente da posição, a solução usual da equação integral para a onda espalhada é obtida via aproximação de Born. Neste artigo apresenta-se duas técnicas alternativas para solução da equação integral contendo um potencial eletrostático. Os métodos usados aqui, são Kerneis arbitrários e a série de Neumann-Born. O resultado, com a ajuda de códigos computacionais, mostra que as duas técnicas são boas comparadas com o método tradicional. A vantagem é que são soluções finitas, que não requer regularização do tipo Podolsky.

Keywords: Espalhamento quântico. Fredholm. Neumann-Born. Modelagem computacional.

Nomenclature

ψ: Wave function
φ: Incident wave

θ: Scattering angle
$f(\theta)$: Scattering Amplitude
$|f(\theta)|^2$: Wave scattering
$V(r)$: Potential energy
Q: Electric charge
r: Distance between two points
k: Wave number
p: Linear momentum of the particle
m: Mass
E: Total Energy
$\hbar$: Reduced Planck constant
$\frac{d\sigma}{d\Omega}$: Differential Shock Session

1. Introduction

Nuclear Physics studies collisions in which a given nucleus is accelerated and launched against a target nucleus, resting in the laboratory frame of reference. The consequence of this collision is given by scattering of these nuclei and measured by a shock session.

An example of the application of this technique to investigate matter is the spontaneous breaking of chiral symmetry, which has a fundamental meaning in understanding the non-perturbation nature of hadron dynamics (Nguyen, 2011). This symmetry breaking around the interaction center, $x = 0$, or scattering center, is little investigated by other techniques for quantum scattering (Sales et. al., 2021). In the literature this problem is known as Endpoint, where several works try to describe the behavior of energy in this scatter center (Nguyen, 2011).

The scattering center studied here, Equation 1, is a central point with an electrostatic potential energy that varies with the inverse of the distance.

$$V(r) = \frac{1}{4\pi}\frac{Q^2}{r} \tag{1}$$

where Q is the electric charge and r is the distance from the center of the potential to the measurement point.

To study the problem of non-relativistic quantum scattering of a particle by a central potential, we use the integral form of the Schrödinger Equation, known as the Lippmann-Schwinger Equation (Sakurai, 2013). In the coordinate representation the Lippmann-Schwinger equation is given by Equation 2,

$$\psi(\vec{r}) = \varphi(\vec{r}) - \frac{m}{2\pi\hbar^2}\int \frac{e^{ik}|\vec{r}-\vec{r}'|}{|\vec{r}-\vec{r}'|} V(\vec{r}')\psi(\vec{r}')d^3r' \tag{2}$$

where $k^2 = \frac{2mE}{\hbar^2}$ is the wave number associated with the incident particle with mass m and the total energy of E, whose time is $p = \hbar k$.

We define the boundary condition for the scattering region away from the potential ($|\vec{r}| > d$), has been: $V(\vec{r}) = 0.$ 0 implies a wave associated with a free particle (Girotto & Sales, 2019)

$$\varphi(\vec{r}) = \frac{1}{2\pi^{3/2}} e^{i\vec{k}.\vec{r}}, \left(\vec{p} = \hbar\vec{k}\right) \tag{3}$$

Which implies a wave associated with a free particle $\varphi(\vec{r})$ that propagates from the z-axis direction, which after iteration through a scattering center, emerges as a spherical wave, as shown in Figure 1 (Griffiths, 2011).

$$\psi(r,\theta) \approx A\left\{e^{ikz} + f(\theta)\frac{e^{ikr}}{r}\right\} \tag{4}$$

where A is a normalization constant, z is the axis of the direction of the incident wave, r distance from the spreader center to the sensor and $f(\theta)$ is the scattering amplitude.

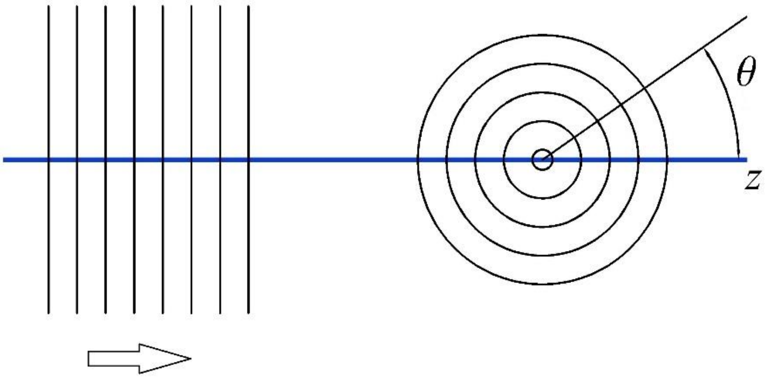


Figure 1 – Wave scattering

The differential shock session is calculated by the absolute square of the scattering amplitude, that is:

$$\frac{d\sigma}{d\Omega} = |f(\theta)|^2 \tag{5}$$

Suppose the desired wave function is calculated at a distance far beyond the scattering center $\vec{r'}$.

$$|\vec{r}| \gg \left|\vec{r'}\right| \tag{6}$$

From this, two approximations can be made:

$$\lim_{r\to\infty}\frac{1}{r-r'} \approx \frac{1}{r}; \left|\vec{r}-\vec{r'}\right|^2 \approx r - \vec{r'}.\hat{r} \tag{7}$$

The Born approximation adopts the assumption that the potential makes no significant change in the scattered wave function, i.e,

$$\psi\left(\vec{r'}\right) \approx \varphi\left(\vec{r'}\right) = e^{ikz'} = e^{i\vec{k'}.\vec{r'}} \tag{8}$$

Using the approximations of Equation 7 and the Born approximation of Equation 8, converting the volume to the spherical $d^3r' = {r'}^2 sin(\phi)d\phi d\gamma dr'$, where $0 \leq \phi \leq \pi$, $0 \leq \gamma \leq 2\pi$ and $0 \leq r' \leq \infty$, get

$$\psi(\vec{r}) = e^{ikz} - \frac{2m}{\hbar^2\Delta k}\frac{e^{ikr}}{r}\int r'V(r')sin(r'\Delta k)dr' \tag{9}$$

Comparing Equation 8 with Equation 4, the scattering amplitude will be

$$f(\theta) = \frac{-2m}{\hbar^2 \Delta k}\int_0^\infty r'V(r')sin(r'\Delta k)dr' \quad (10)$$

where $\Delta k = 2ksin\frac{\theta}{2}$ (Griffiths, 2019).

For a validation of Born's approximation the data from Geiger's experiment is used -Marsden (1913). The Geiger-Marsden experiment consisted of firing alpha particles generated by the radioactive decay of radon towards a very thin gold leaf in an evacuated chamber, the experiment showed the existence of the atomic nucleus.

For Theoretical scattering is used equation 10 with the potential $V(r) = Z_1 Z_2 \frac{e^2}{r}$, where Z_1 is the atomic number of gold ($Z_1 = 79$), Z_2 is the atomic number of the alpha particle ($Z_2 = 2$) and the energy being equal to 5.69 MeV, which is the same Geiger-Marsden data.

Figure 2 shows that the theoretical scattering in the Born approach passes through the Geiger-Marsden experimental points (Geiger & Marsden, 1913), thus validating the use of the Born approximation and equation 10, where $\frac{d\sigma}{d\Omega}$ it is given by Equation 5.

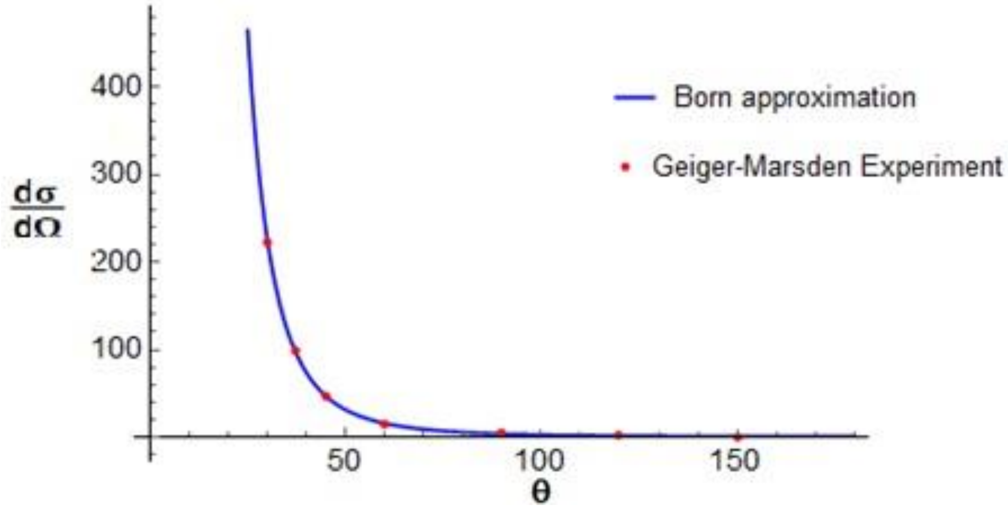


Figure 2 – Theoretical scattering, via Born approximation, and the observed data.

This article shows two alternative techniques for non-relativistic quantum scattering that differ from the traditional method known as Born approximation. The first method solves Integral Equations by the technique of arbitrary kernels by the Fredholm determinant.

The second seeks solution by the linear operator called the Neumann – Born series. The technique used for the solutions of the two methods are computational codes developed in the Wolfram Mathematics environment.

2. Fredholm Integral equation

2.1 Arbitrary kernel method

An integral equation is called an equation that contains the unknown function under the integration sign, as for example the second-kind Fredholm equation (Bassalo & Cattani, 2012):

$$u(x) = f(x) + \lambda \int_a^b K(x,t)u(t)dt \tag{11}$$

where $K(x,t)$ and $f(x)$ are functions known to $a \leq x \leq b$ e $a \leq t \leq b$. This equation was solved by Fredholm in 1900, using the trick of replacing the integral of Equation 11 with the corresponding Riemann sum.

We divide the interval (a,b) in n equal intervals (Arfken & Weber, 2007):

$$\frac{b-a}{n} = \Delta x \\ = \Delta t \tag{12}$$

For a generic point $x_h(t_h)$ implies in the equation $\Delta x = \frac{x_h(t_h)-a}{h}$ when $x_h = b$,this implies $h = n$, leading to Equation 12. Therefore, we can define that the value of $x_h(t_h)$ for any point within the region is $x_h(t_h) = a + h\Delta x(\Delta t)$. To make writing easier, we'll set the annotations to $f_h = f(x_h)$, to $u_q = u(x_h(t_h))$ and to $u_q = u(x_h(t_h))$, where $h, p, q = 1,2,3, \dots, n$.

Thus, replacing the integral of Equation 11 by a summation, it will come:

$$u(x) = f(x) + \lambda \sum_{q=1}^{n} K(x, t_q) u_q \, \Delta t \tag{13}$$

replacing in Equation 13 the variable x by x_p, one obtains a system of n first degree linear equations of unknown functions $u_1, u_2, \dots, u_n$. thus

$$u_p = f_p + \lambda \sum_{q=1}^{n} K_{pq} \, u_q \Delta t \tag{14}$$

to obtain the solution of the system given by Equation 14 is put:

$$u_p = \sum_{q=1}^{n} \delta_{pq} \, u_q \tag{15}$$

which, taken in Equation 14, will give:

$$\sum_{q=1}^{n} [\delta_{pq} - \lambda K_{pq} \Delta t] u_q = f_p \tag{16}$$

which is a system of equations for$p = 1,2, \dots, n$ e $q = 1,2, \dots, n$.

Using Cramer's rule (Krasnov, 1981) to solve this system, we will have:

$$u_q = \frac{\Delta_{nq}(\lambda)}{\Delta_n(\lambda)} \tag{17}$$

where $\Delta_n(\lambda)$ it is the determinant of the coefficients of the unknowns and $\Delta_{nq}(\lambda)$ It is the determinant of the coefficients of the unknowns andu_q by column of independent terms f_p. The determinant $\Delta_n(\lambda)$ it is given by:

$$\varDelta_n(\lambda) = \begin{bmatrix} 1 - \lambda K_{11}\varDelta t & -\lambda K_{12}\varDelta t & \dots & -\lambda K_{1n}\varDelta t \\ -\lambda K_{21}\varDelta t & 1 - \lambda K_{22}\varDelta t & \dots & -\lambda K_{2n}\varDelta t \\ \vdots & \vdots & \ddots & \vdots \\ -\lambda K_{n1}\varDelta t & -\lambda K_{n2}\varDelta t & \dots & 1 - \lambda K_{nn}\varDelta t \end{bmatrix} \tag{18}$$

where k_{ij} corresponds to i-line $(i = 1,2, \dots, n)$ and j-column $(j = 1,2, \dots, n)$. on the main diagonal, $\delta_{pq} = 1$ to $p = q$ and $\delta_{pq} = 0$ to $p \neq q$.

Applying to Equation 18 the decomposition formula of a determinant (Smirnov, 1975) will come:

$$\varDelta_n(\lambda) = 1 - \frac{\lambda}{1!}\sum_{p_1=1}^{n} K_{p_1p_1}\,\varDelta t + \frac{\lambda^2}{2!}\sum_{p_1=1}^{n}\sum_{p_2=1}^{n} \begin{Bmatrix} K_{p_1p_1} & K_{p_1p_2} \\ K_{p_2p_1} & k_{p_2p_2} \end{Bmatrix} (\varDelta t)^2 + \cdots$$
$$+ (-1)^n \frac{\lambda^n}{n!}\sum_{p_1=1}^{n}\sum_{p_2=1}^{n} \dots \sum_{p_n=1}^{n} \begin{bmatrix} K_{p_1p_1} & K_{p_1p_2} & \dots & K_{p_1p_n} \\ K_{p_2p_1} & K_{p_2p_2} & \dots & K_{p_2p_n} \\ \vdots & \vdots & \ddots & \vdots \\ K_{p_np_1} & K_{p_np_2} & \dots & K_{p_np_n} \end{bmatrix} (\varDelta t)^n \tag{19}$$

now let us consider successively the terms of the second member of Equation 19. Now, the Riemann sums of this equation can be replaced by integrals at the limit $n \to \infty$. So, we will have for the second term and for the third term, respectively:

$$\sum_{p_1=1}^{n} K_{p_1p_1}\,\varDelta t = \sum_{t=1}^{n} K(t_1, t_1)\varDelta t = \int_a^b K(t_1, t_1)dt \tag{20}$$

$$\sum_{p_1=1}^{n}\sum_{p_2=1}^{n} \begin{bmatrix} K_{p_1p_1} & K_{p_1p_2} \\ K_{p_2p_1} & K_{p_2p_2} \end{bmatrix} (\varDelta t)^2 = \int_a^b \int_a^b \begin{bmatrix} K(t_1, t_1) & K(t_1, t_2) \\ K(t_2, t_1) & K(t_2, t_2) \end{bmatrix} dt_1\, dt_2 \tag{21}$$

and so on. Thus, Equation 19 will be:

$$\varDelta_n(\lambda) = 1 + \sum_{n=1}^{\infty} (-1)^n \frac{\lambda^n}{n!} d_n \tag{22}$$

where

$$d_n = \int_a^b \int_a^b \dots \int_a^b \begin{bmatrix} K(t_1, t_1) & K(t_1, t_2) & \dots & K(t_1, t_n) \\ K(t_2, t_1) & K(t_2, t_2) & \dots & K(t_2, t_n) \\ \vdots & \vdots & \ddots & \vdots \\ K(t_n, t_1) & K(t_n, t_2) & \dots & K(t_n, t_n) \end{bmatrix} dt_1\, dt_2 \dots dt_n \tag{23}$$

The value of the determinant $\varDelta_{nq}(\lambda)$ of Equation 17 was obtained by Fredholm (1900), with a calculation that involves a lot of algebraic manipulation, which is why we will only present the result (Krasnov, 1981). Thus:

$$\varDelta_{nq}(\lambda) \equiv \varDelta(x,t;\lambda) = K(x,t) + \sum_{n=1}^{\infty} (-1)^n \frac{\lambda^n}{n!} d_n(x,t) \tag{24}$$

where

$$d_n(x,t) = \int_a^b \int_a^b \; ... \int_a^b \begin{bmatrix} K(x,t) & K(x,t_1) & K(x,t_2) & ... & K(x,t_n) \\ K(t_1,t) & K(t_1,t_1) & K(t_1,t_2) & ... & K(t_1,t_n) \\ K(t_2,t) & K(t_2,t_1) & K(t_2,t_2) & ... & K(t_2,t_n) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ K(t_n,t) & K(t_n,t_1) & K(t_n,t_2) & ... & K(t_n,t_n) \end{bmatrix} dt_1\, dt_2 \, ...\, dt_n \tag{25}$$

Thus, according to Fredholm, the solution of Equation 11 will be given by:

$$u(x) = f(x) + \lambda \int_a^b \frac{\varDelta(x,t;\lambda)}{\varDelta(\lambda)} f(t)dt \tag{26}$$

where $\varDelta(x,t;\lambda)$ e $\varDelta(\lambda)$ are data, respectively, by Equations 22, 23 and Equations 24, 25.

2.2 Neumann-Born Series

Another example of an integral equation is the Neumann-Born series which is a method for solving Fredholm Integral Equations in which it uses successive approximations (Arfken & Weber, 2007). Therefore, the structure of the solution of Equation 11 will be:

$$u_n(x) = f(x) + \lambda \int_a^b K(x,t)u_{n-1}(t)dt \tag{27}$$

Using Equation 27 successively

$$u_1(X) = f(x) + \lambda \int_a^b K(x,t)u_0(t)dt \tag{28}$$

$$u_2(x) = f(x) + \lambda \int_a^b K(x,t)u_1(x)dt \tag{29}$$

Replacing Equation 28 into Equation 29

$$u_2(x) = f(x) + \lambda \int_a^b K(x,t) \left[f(t) + \lambda \int_a^b K(t,t_1)u_0(t_1)dt1 \right] dt$$

$$\begin{aligned} u_2(x) = f(x) + \lambda \int_a^b K(x,t)f(t)dt \\ + \lambda^2 \int_a^b K(x,t) \int_a^b K(t,t_1)u_0(x)dt_1\, dt \end{aligned} \tag{30}$$

and so on to u_3, u_4 e u_{n-1}.

Introducing the linear integral operator

$$\kappa g(x) \equiv \int_a^b K(x,t)g(t)dt \tag{31}$$

Thus, Equations 28, 30 can be rewritten

$$u_1(x) = f(x) + \lambda\kappa u_0(x)$$

$$u_2(x) = f(x) + \lambda\kappa f(x) + \lambda^2\kappa^2 u_0(x)$$

Generalizing it will come:

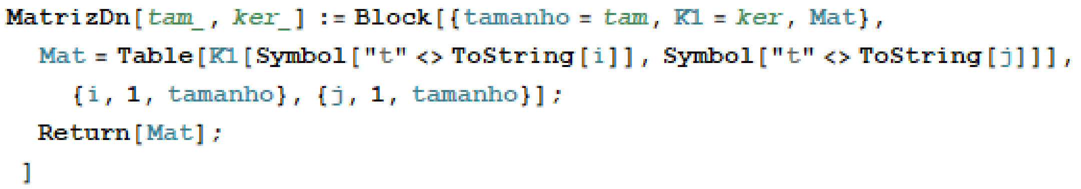
$$u(x) = f(x) + \sum_{n=1}^{\infty} \lambda^n\, \kappa^n f(x) \tag{32}$$

3. Algorithms for Integral Equation

3.1 Implementation of the arbitrary kernel method

The calculations for solving equation 11 are long because of the presence of the determinants that are inserted into the integrals of the equations 23 e 24. That's why a code was developed in Wolfram Mathematics to solve Fredholm's integral equations through the arbitrary kernel method, where the code can be divided into six blocks (Sales, J.H. et. al.,2021).

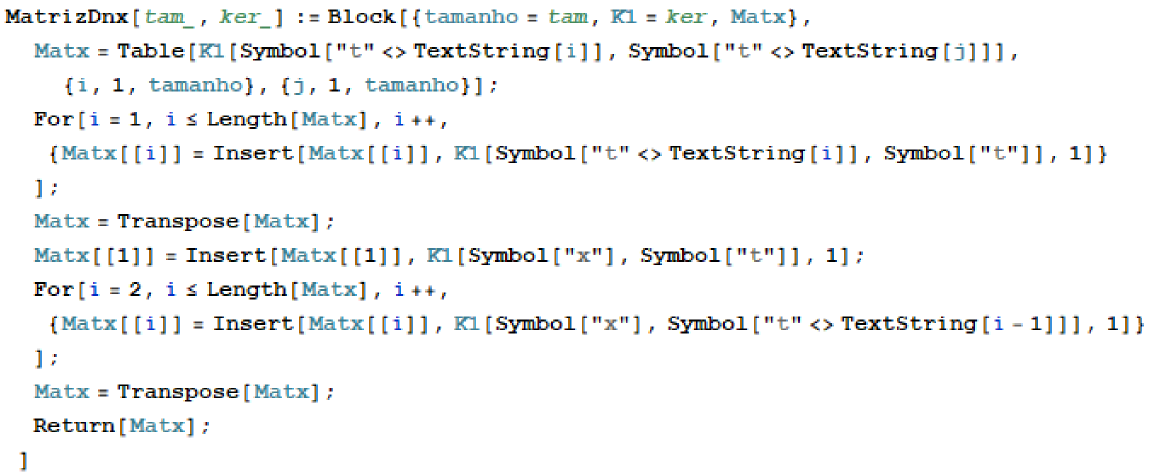
Figure 3 is the function MatrizDn which has as input parameter the size n of the Matrix and the expression of the kernel $K(x,t)$ and this way builds the matrix that will be used to calculate the determinants of d_n in the Equations 22 e 23.



```
MatrizDn[tam_, ker_] := Block[{tamanho = tam, K1 = ker, Mat},
  Mat = Table[K1[Symbol["t" <> ToString[i]], Symbol["t" <> ToString[j]]],
    {i, 1, tamanho}, {j, 1, tamanho}];
  Return[Mat];
 ]
```

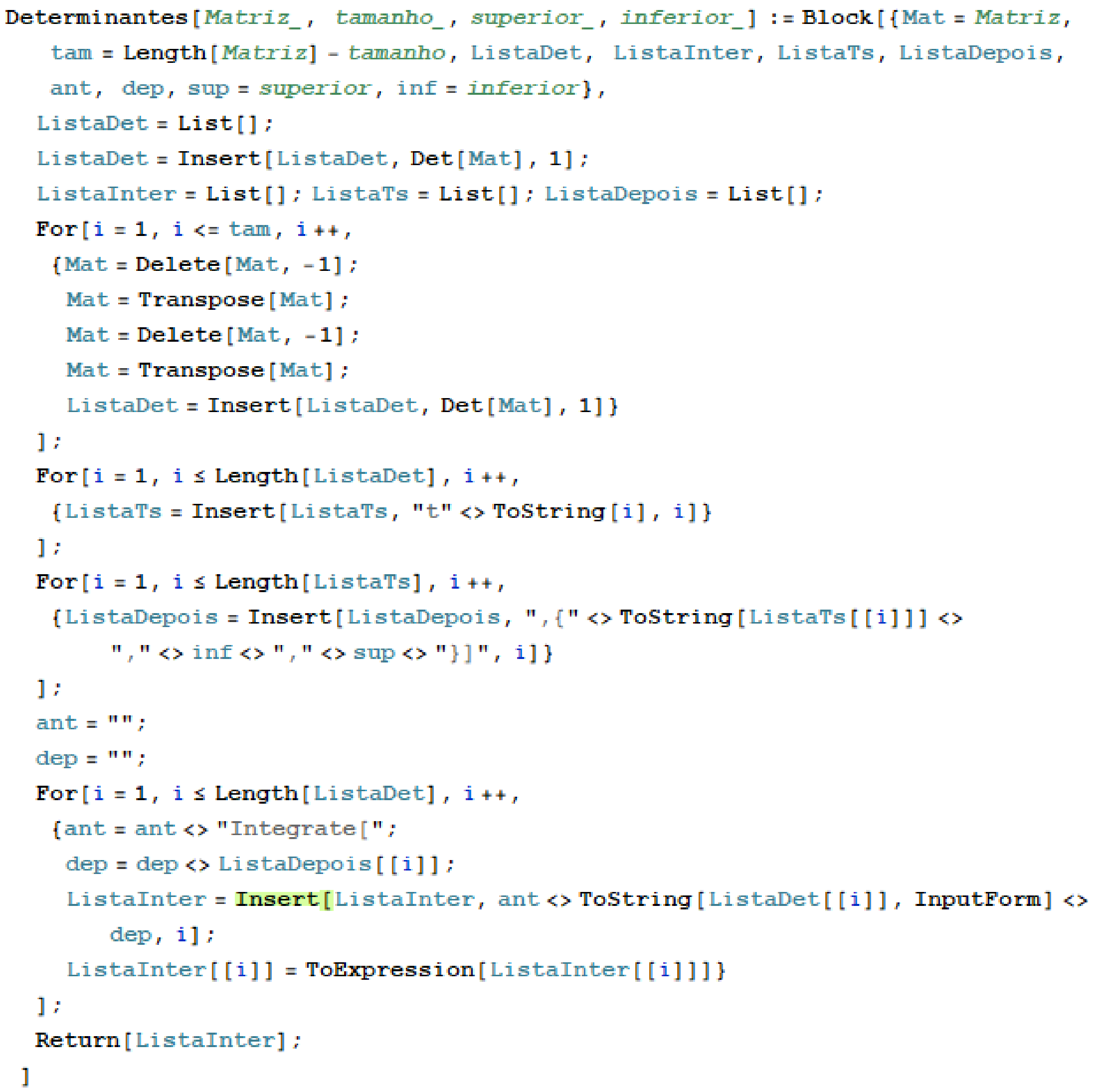
Figure 3- First block of the arbitrary kernel method implementation.

Figure 4 is the function MatrizDnx which has as input parameter the size n of the Matrix and the expression of the kernel $K(x,t)$ and this way builds the matrix that will be used to calculate the determinants of $d_n(x,t)$ in the Equations 24 e 25. Figure 5 is the Determinants function whose input parameter is the matrix, the smallest size for the sub-matrix, the value of the upper and lower limit of the integral. With this the block calculates the matrix determinants, starting with the matrix determinant calculation. $n \times n$ and thus, calculating the determinants of the sub-matrices, built from the removal of the last row and column. With the calculated determinants, the block solves the integration's and thus obtains the values of d_n or $d_n(x,t)$ in the Equations 23 e 24.



```
MatrizDnx[tam_, ker_] := Block[{tamanho = tam, K1 = ker, Matx},
  Matx = Table[K1[Symbol["t" <> TextString[i]], Symbol["t" <> TextString[j]]],
    {i, 1, tamanho}, {j, 1, tamanho}];
  For[i = 1, i ≤ Length[Matx], i ++,
   {Matx[[i]] = Insert[Matx[[i]], K1[Symbol["t" <> TextString[i]], Symbol["t"]], 1]}
  ];
  Matx = Transpose[Matx];
  Matx[[1]] = Insert[Matx[[1]], K1[Symbol["x"], Symbol["t"]], 1];
  For[i = 2, i ≤ Length[Matx], i ++,
   {Matx[[i]] = Insert[Matx[[i]], K1[Symbol["x"], Symbol["t" <> TextString[i - 1]]], 1]}
  ];
  Matx = Transpose[Matx];
  Return[Matx];
 ]
```

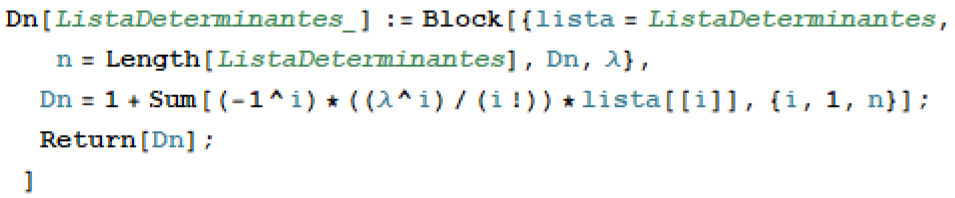
Figure 4- Second block of the implementation of the arbitrary kernel method.



```
Determinantes[Matriz_, tamanho_, superior_, inferior_] := Block[{Mat = Matriz,
   tam = Length[Matriz] - tamanho, ListaDet, ListaInter, ListaTs, ListaDepois,
   ant, dep, sup = superior, inf = inferior},
  ListaDet = List[];
  ListaDet = Insert[ListaDet, Det[Mat], 1];
  ListaInter = List[]; ListaTs = List[]; ListaDepois = List[];
  For[i = 1, i <= tam, i ++,
   {Mat = Delete[Mat, -1];
    Mat = Transpose[Mat];
    Mat = Delete[Mat, -1];
    Mat = Transpose[Mat];
    ListaDet = Insert[ListaDet, Det[Mat], 1]}
  ];
  For[i = 1, i ≤ Length[ListaDet], i ++,
   {ListaTs = Insert[ListaTs, "t" <> ToString[i], i]}
  ];
  For[i = 1, i ≤ Length[ListaTs], i ++,
   {ListaDepois = Insert[ListaDepois, ",{" <> ToString[ListaTs[[i]]] <>
       "," <> inf <> "," <> sup <> "}]", i]}
  ];
  ant = "";
  dep = "";
  For[i = 1, i ≤ Length[ListaDet], i ++,
   {ant = ant <> "Integrate[";
    dep = dep <> ListaDepois[[i]];
    ListaInter = Insert[ListaInter, ant <> ToString[ListaDet[[i]], InputForm] <>
       dep, i];
    ListaInter[[i]] = ToExpression[ListaInter[[i]]]}
  ];
  Return[ListaInter];
 ]
```

Figure 5- Third block of the arbitrary kernel method implementation.

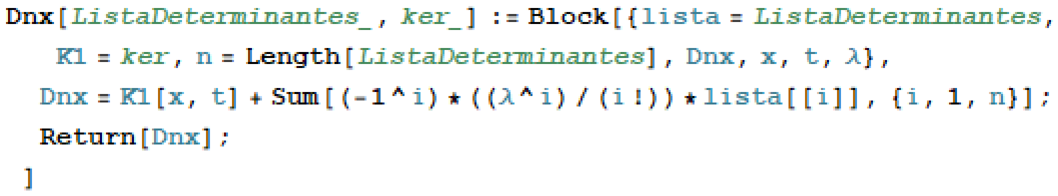
Figure 6 is the function Dn which has as input parameter the list containing the values of d_n and accordingly calculates the value of $\Delta(\lambda)$ from the Equation 22.



```
Dn[ListaDeterminantes_] := Block[{lista = ListaDeterminantes,
   n = Length[ListaDeterminantes], Dn, λ},
  Dn = 1 + Sum[(-1^i) * ((λ^i) / (i!)) * lista[[i]], {i, 1, n}];
  Return[Dn];
 ]
```

Figure 6- Fourth block of the implementation of the arbitrary kernel method.

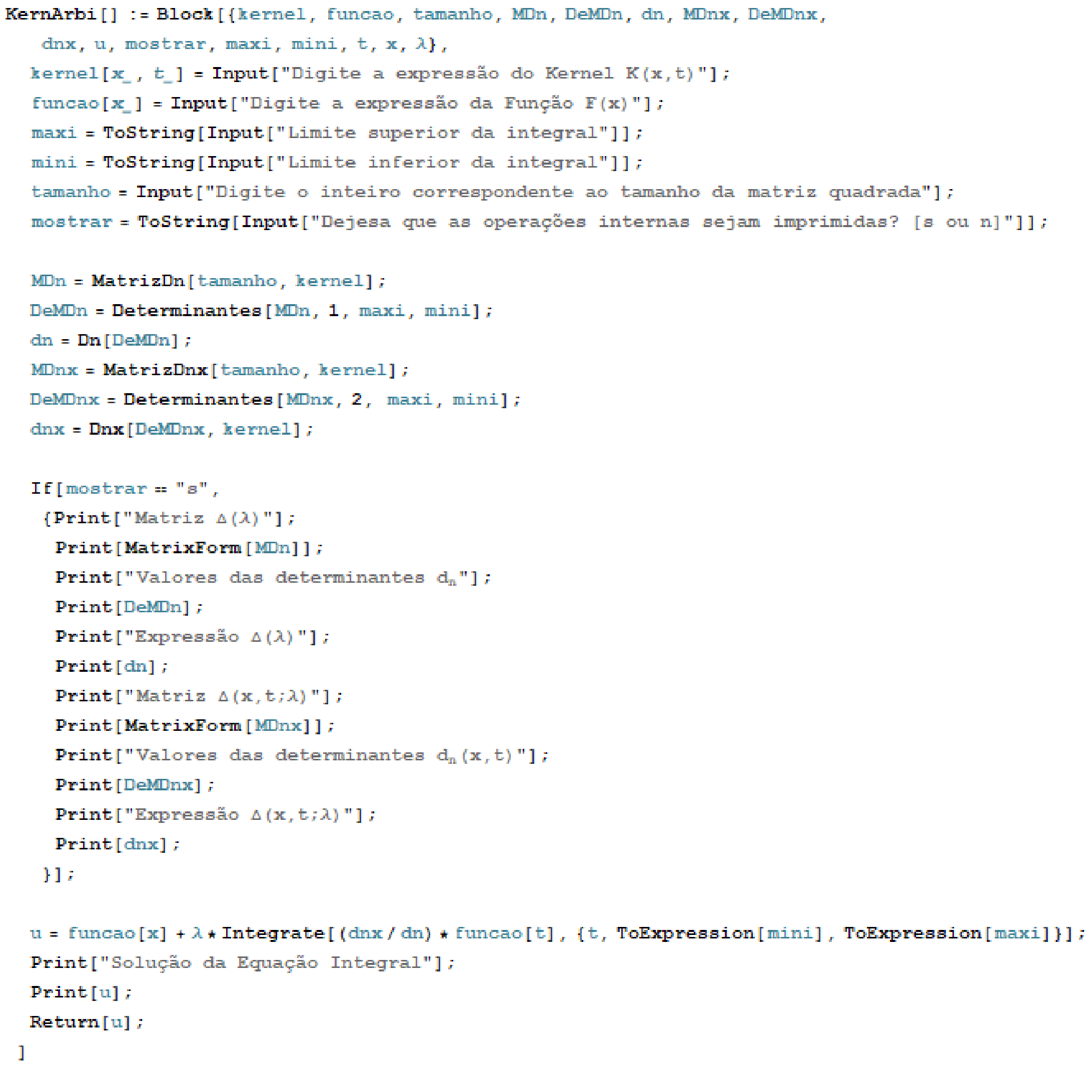
Figure 7 is the function Dnx which has as input parameter the list containing the values of $d_n(x,t)$ and the kernel $K(x,t)$ and like this calculates the value of $\Delta(x,t;\lambda)$ from the Equation 24.



```
Dnx[ListaDeterminantes_, ker_] := Block[{lista = ListaDeterminantes,
   K1 = ker, n = Length[ListaDeterminantes], Dnx, x, t, λ},
  Dnx = K1[x, t] + Sum[(-1^i) * ((λ^i) / (i!)) * lista[[i]], {i, 1, n}];
  Return[Dnx];
 ]
```

Figure 7- Fifth block of the implementation of the arbitrary kernel method.

Figure 8 is the function KernArbi which aims to make life easier for the programmer, instead of the user using the five other separate functions one at a time, *KernArbi* jjoins them all in sequence to calculate and returns to the solution of the integral equation.



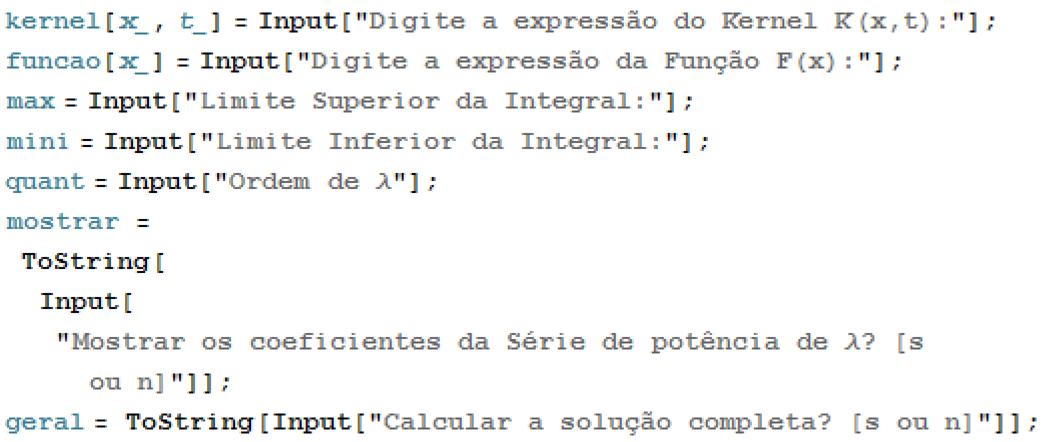
```
KernArbi[] := Block[{kernel, funcao, tamanho, MDn, DeMDn, dn, MDnx, DeMDnx,
   dnx, u, mostrar, maxi, mini, t, x, λ},
  kernel[x_, t_] = Input["Digite a expressão do Kernel K(x,t)"];
  funcao[x_] = Input["Digite a expressão da Função F(x)"];
  maxi = ToString[Input["Limite superior da integral"]];
  mini = ToString[Input["Limite inferior da integral"]];
  tamanho = Input["Digite o inteiro correspondente ao tamanho da matriz quadrada"];
  mostrar = ToString[Input["Dejesa que as operações internas sejam imprimidas? [s ou n]"]];

  MDn = MatrizDn[tamanho, kernel];
  DeMDn = Determinantes[MDn, 1, maxi, mini];
  dn = Dn[DeMDn];
  MDnx = MatrizDnx[tamanho, kernel];
  DeMDnx = Determinantes[MDnx, 2, maxi, mini];
  dnx = Dnx[DeMDnx, kernel];

  If[mostrar == "s",
   {Print["Matriz Δ(λ)"];
    Print[MatrixForm[MDn]];
    Print["Valores das determinantes d_n"];
    Print[DeMDn];
    Print["Expressão Δ(λ)"];
    Print[dn];
    Print["Matriz Δ(x,t;λ)"];
    Print[MatrixForm[MDnx]];
    Print["Valores das determinantes d_n(x,t)"];
    Print[DeMDnx];
    Print["Expressão Δ(x,t;λ)"];
    Print[dnx];
   }];

  u = funcao[x] + λ * Integrate[(dnx / dn) * funcao[t], {t, ToExpression[mini], ToExpression[maxi]}];
  Print["Solução da Equação Integral"];
  Print[u];
  Return[u];
 ]
```

Figure 8- Sixth block of the implementation of the arbitrary kernel method.

3.2 Implementation of the Neumann-Born series method

Here, the difficulty in solving Equation 11 is the series orders, which become increasingly difficult because of the iterated integrals. which means a code was developed in Wolfram Mathematics to solve the Fredholm integral equations using the Neumann-Born series method.

Figure 9 is the first part of the code and aims to receive the user inputs, which consists of the kernel expression, expression of the incident wave function, the upper and lower limit of the integral, the order of the polynomial in whether the coefficients of the polynomials should be printed on the screen and finally whether the function should calculate a general expression for the polynomial.

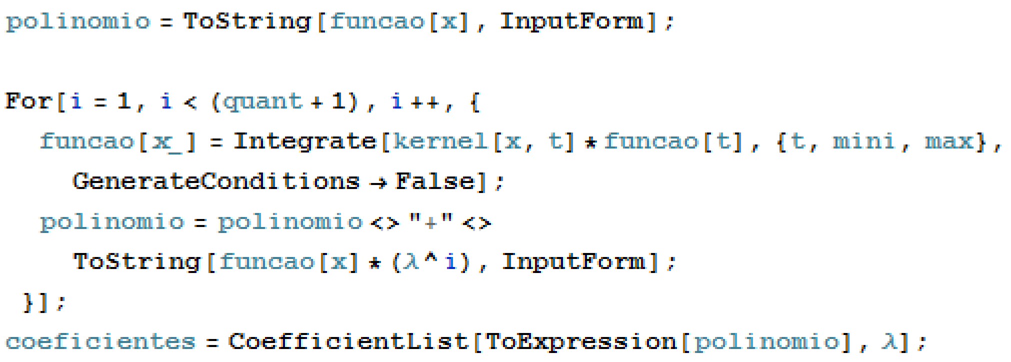


```
kernel[x_, t_] = Input["Digite a expressão do Kernel K(x,t):"];
funcao[x_] = Input["Digite a expressão da Função F(x):"];
max = Input["Limite Superior da Integral:"];
mini = Input["Limite Inferior da Integral:"];
quant = Input["Ordem de λ"];
mostrar =
 ToString[
  Input[
   "Mostrar os coeficientes da Série de potência de λ? [s
     ou n]"]];
geral = ToString[Input["Calcular a solução completa? [s ou n]"]];
```

Figure 9- First block of the implementation of the Neumann-Born method.

Figure 10 is the second part of the code consisting of the calculation, where the *polynomial* variable will store the polynomial in the form of a *string.*

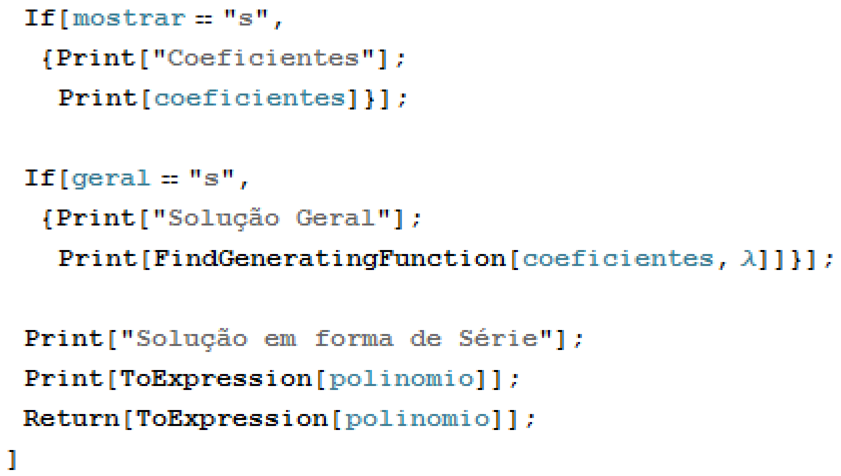
The repeat loop *For* it is where the successive approximations will occur, where the value of the variable *function* is updated with the calculation of the Integral of the multiplication of the *kernel* with the old value of the *function*, so the value of *function* is multiplied λ with the order of the iteration and transformed to *string*, making the concatenation with the string of the *polynomial.*



```
polinomio = ToString[funcao[x], InputForm];

For[i = 1, i < (quant + 1), i ++, {
  funcao[x_] = Integrate[kernel[x, t] * funcao[t], {t, mini, max},
    GenerateConditions → False];
  polinomio = polinomio <> "+" <>
    ToString[funcao[x] * (λ ^ i), InputForm];
 }];
coeficientes = CoefficientList[ToExpression[polinomio], λ];
```

Figure 10 - Second block of the implementation of the Neumann-Born method.

Figure 11 is the final part of the code that is responsible for printing the results, highlighting the Wolfram Mathematics *FindGeneratingFunction* Command, which is a command that receives a series and tries to calculate the expression that generates the series.



```
    If[mostrar == "s",
     {Print["Coeficientes"];
      Print[coeficientes]}];

    If[geral == "s",
     {Print["Solução Geral"];
      Print[FindGeneratingFunction[coeficientes, λ]]}];

    Print["Solução em forma de Série"];
    Print[ToExpression[polinomio]];
    Return[ToExpression[polinomio]];
  ]
```

Figure 11- Third block of the implementation of the Neumann-Born method.

4. Results

4.1 Quantum scattering using arbitrary kernel methods

This section shows how the codes work when you have an electrostatic potential like:

$$V(r') = \frac{1}{4\pi}\frac{Q^2}{r'} \tag{33}$$

whose general solution of Equation 2 and with the approximations of Equation 7, is given by

$$\psi(\vec{r}) = e^{i\vec{k'}.\vec{r'}} - \frac{mQ^2}{8\pi^2\hbar^2}\frac{e^{ikr}}{r}\int\frac{e^{-i\vec{k}.\vec{r'}}}{r'}\psi\left(\vec{r'}\right)d^3\,r' \tag{34}$$

where Q is the charge of the particles, m is the mass of the particle shot towards the scattering center, r is the range distance of the potential and r' is the distance from the spreader center to the sensor.

Instead of using the Born approximation, Equation 8, an alternative hypothesis is presented, where the scattered wave number has a small perturbation, ie,

$$\psi\left(\vec{r'}\right) \approx e^{i\vec{r'}\left(\vec{k'}+\Delta\vec{\kappa'}\right)} \approx e^{i\vec{r'}\cdot\vec{k'}}e^{i\vec{r}'\Delta\vec{k}'} \tag{35}$$

where $\Delta\vec{\kappa'}$ is the variation that k suffers when the particle collides with electrostatic potential energy.

Defining

$$\psi'(\vec{r}) \equiv e^{i\Delta\vec{k}\cdot\vec{r}} \tag{36}$$

then

$$\psi\left(\vec{r'}\right) \approx e^{i\vec{k'}\cdot\vec{r'}}\psi'\left(\vec{r'}\right) \tag{37}$$

This hypothesis presents a particular case, where the perturbation tending to zero implies the Born approximation, Equation 8.

Using the approximation of Equation 37 to Equation 34 and converting the volume d^3r' for the spherical volume and calculating the integral in $d\phi$. The integral equation 34 will be

$$\psi(\vec{r}) = e^{ik'r'} - \frac{mQ^2}{4\pi\hbar^2 k sen(\theta/2)} \int \frac{e^{ikr}}{r} sen[2kr' sen(\theta/2)] \psi'(\vec{r'}) dr' \tag{38}$$

Remembering that $\vec{k'}$ and $\vec{r'}$ are vectors and doing vector scalar multiplication $\vec{k'} \cdot \vec{r'} = kr'$, where the module of $\vec{k'}$ is equal to $\vec{k}$.

With Equation 38 we can feed the arbitrary kernel method code, which receives the values from Table 1 as an input parameter.

Table 1- Input parameters for arbitrary kernels method on Coulomb potential

input parameter	Value
Kernel Expression	$e^{ikr} sin[2kr' sin(\theta/2)]/r$
Incident Wave Expression	$e^{ikr'}$
Upper Limit of Integral	10
Lower Limit of Integral	0
Matrix Size	5
Print Operations	Yes

For equation 38, the code returns as a result.

$$\psi(\vec{r}) = e^{ikr'} + \frac{e^{ikr}}{r} \cdot \\ \cdot \left\{ \frac{4\lambda \left\{ 2i \left[-1 + e^{10ik} cos\left[20k sin\left(\frac{\theta}{2}\right)\right] \right] sin\left[\frac{\theta}{2}\right] + e^{10ik} sin\left[20k sin\left(\frac{\theta}{2}\right)\right] \right\}}{k[-1 + 2cos(\theta)][4i + \lambda A]} \right\} \tag{39}$$

where

$$A = \left\{ 2Ei\left[-10ik\left(2sin\left(\frac{\theta}{2}\right) - 1\right)\right] - 2Ei\left[10i\left(2sin\left(\frac{\theta}{2}\right)k + k\right)\right] + \\ + log\left[ik\left(2sin\left(\frac{\theta}{2}\right) + 1\right)\right] - log\left[\frac{-i}{2k sin\left(\frac{\theta}{2}\right) + k}\right] + \\ - log\left[ik\left(1 - 2sin\left(\frac{\theta}{2}\right)\right)\right] + log\left[\frac{-i}{k - 2k sin\left(\frac{\theta}{2}\right)}\right] \right\} \tag{40}$$

being the value of λ the terms that multiply the integral (38)

$$\lambda = \frac{-mQ^2}{4k\pi\hbar^2 sin(\theta/2)} \tag{41}$$

Comparing the result of Equation 39 with the general solution of the Schrödinger equation, Equation 4, the scattering amplitude value will be:

$$f(\theta)=\frac{4\lambda\left\{2i\left[-1+e^{10ik}cos\left[20ksin\left(\frac{\theta}{2}\right)\right]\right]sin\left[\frac{\theta}{2}\right]+e^{10ik}sin\left[20ksin\left(\frac{\theta}{2}\right)\right]\right\}}{k[-1+2cos(\theta)][4i+\lambda A]} \tag{42}$$

For an initial study, we consider that the mass m be equal to 1, the charge Q be equal to 1 and that Planck's reduced constant $\hbar$ be equal to 1. With these values we will have $\lambda=\frac{-1}{4k\pi sin(\theta/2)}$ and a non-relativistic limit set for $k=\frac{1}{\hbar}\sqrt{2mE}$, because for relativistic energy $E=mc^2=1$, which implies $k\approx 1.4$. Therefore, for non-relativistic values k must vary between 0 and 1.4.

In Figure 12 shows the scattering as a function of the angle θ and the wave number k, a range is observed between 0.3 a 0.5 to k that intercepted the two graphics. Therefore, is chosen $k=0.44$ to describe a good behavior of the Fredholm method with Born approximation for scattering.

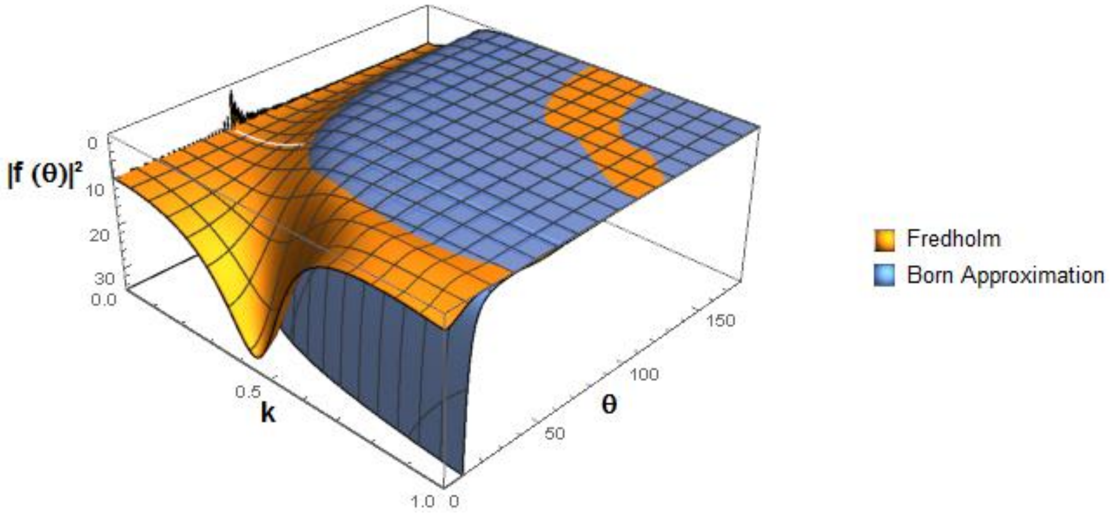


Figure 12- Fredholm and Born scattering in function of k and angle θ.

Figure 13 is the scattering of the Fredholm method and the Born method, where k chosen is equal to 0.44 and the scattering angle θ is ranging from 0° to 180°.

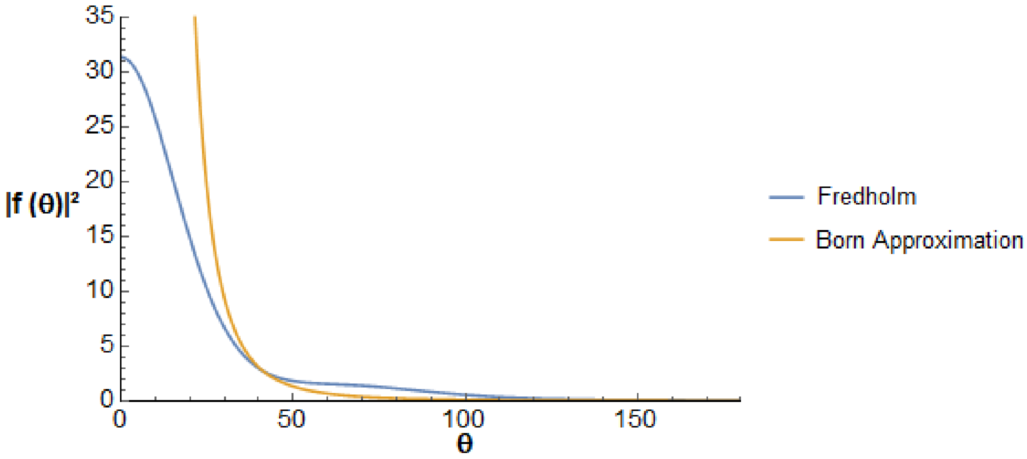


Figure 13- Fredholm scattering compared to Born scattering.

The scattering via Fredholm has no singularities for angle close to zero, as seen in Figure 13.

4.2 Quantum scattering using Neumann-Born series methods

In Equation 38 the Neumann-Born series method is used, which receives the input parameters:

Table 2- Input parameter values for the Neumann-Born method in Coulomb potential

Input Parameter	Value
Kernel Expression	$e^{ikr}sin[2kr'sin(\theta/2)]/r$
Incident Wave Expression	$e^{ikr'}$
Upper Limit of Integral	10
Lower Limit of Integral	0
Grade of Series λ	5
Print Coefficients	No
Generating Function	No

For Equation 38, the code returns the result

$$\psi(\vec{r}) = e^{ikr'} + \frac{e^{ikr}}{r}\frac{\lambda}{k[2cos(\theta)-1]}\left\{[i \times C_0] + \left[\frac{\lambda \times C_0 \times A}{4}\right] + \left[\frac{i\lambda^2 \times C_0 \times A^2}{16}\right] - \left[\frac{\lambda^3 \times C_0 \times A^2 \times C_1}{32}\right] + \left[\frac{i\lambda^4 \times C_0 \times A^2 \times C_1^2}{64}\right]\right\} \quad (43)$$

where

$$C_0 = \left\{e^{10ik}sin\left[20ksin\left(\frac{\theta}{2}\right)\right] + 2isin\left(\frac{\theta}{2}\right)\left[-1 + e^{10ik}cos\left(20ksin\left(\frac{\theta}{2}\right)\right)\right]\right\}$$

$$C_1 = \left\{-\mathrm{Ci}\left[10k\left[2sin\left(\frac{\theta}{2}\right)+1\right]\right] + \mathrm{Ci}\left[10k\left[1-2sin\left(\frac{\theta}{2}\right)\right]\right] - i\mathrm{Si}\left[10k\left[2sin\left(\frac{\theta}{2}\right)+1\right]\right] + i\mathrm{Si}\left[10k\left[1-2sin\left(\frac{\theta}{2}\right)\right]\right] + log\left[2ksin\left(\frac{\theta}{2}\right)+k\right] - log\left[k-2ksin\left(\frac{\theta}{2}\right)\right]\right\}$$

being the value of λ the terms that multiply the integral 41.

Comparing the result of Equation 43 with the general solution of Schrödinger's equation 4, the scattering amplitude value will be:

$$f(\theta) = \frac{\lambda}{k[2cos(\theta)-1]}\left\{[i \times C_0] + \left[\frac{\lambda \times C_0 \times A}{4}\right] + \left[\frac{i\lambda^2 \times C_0 \times A^2}{16}\right] - \left[\frac{\lambda^3 \times C_0 \times A^2 \times C_1}{32}\right] + \left[\frac{i\lambda^4 \times C_0 \times A^2 \times C_1^2}{64}\right]\right\} \quad (44)$$

where **A** is defined in the Equation 40.

For a consistent analysis we will use the same values for the mass. $m = 1$, the load $Q = 1$ and Planck's constant $\hbar = 1$. Therefore, we can compare the result of scattering 44 with the usual technique (Born approximation), using solution 9 for the electrostatic potential.

Figure 14 shows the scattering given by the Born approximation and via the Neumann-Born method, with the Coulomb potential, where the wave number k is equal to 0.44 and the scattering angle θ is ranging from 0° to 180°.

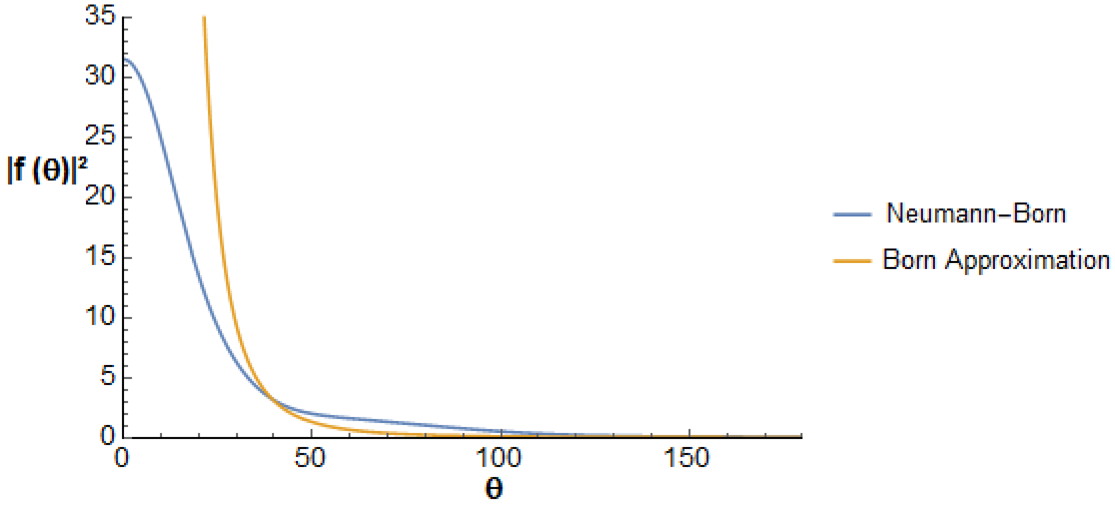


Figure 14- Neumann-Born scattering compared with Born scattering.

Figure 15 shows the scattering of the Born approximation and the three-series scattering of the Neumann-Born method, where the orders are from 1°, 5° and 9°, the wave number k is equal to 0.44 and the scattering angle θ is ranging from 0° to 180°.

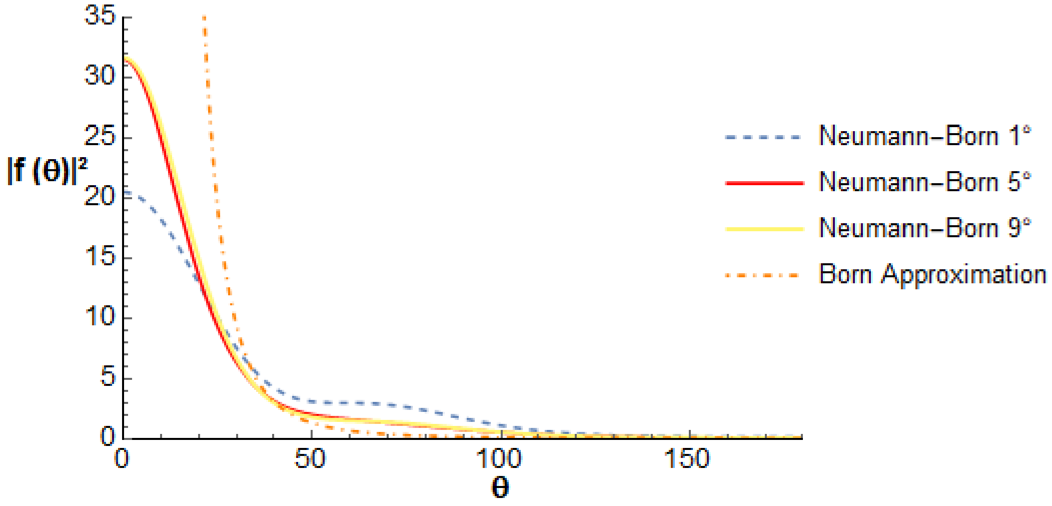


Figure 15- Scattering via Neumann-Born and Born approximation.

A difference between the methods is that in the Neumann-Born method there isn't singularity when $\theta \to 0$, which doesn't happen in the Born approximation and order is observed convergence to 9°.

4.3 Comparative between methods

Two methods were used to calculate the scattering amplitude through the Schrödinger integral equation 2. Figure 16 shows the scattering of the arbitrary kernel method (Fredholm) and the Neumann-Born method for the same conditions used in the previous cases, where the mass $m = 1$, load $Q = 1$ and the reduced Planck constant $\hbar = 1$ for Coulomb's potential. The Neumann-Born method was used to generate three series, one of the first grade, one of the fifth and one of the ninth degree, to compare the result of series with a higher grade. In Figure 16 it shows that increasing the order of the Neumann-Born series, it approaches the Fredholm method.

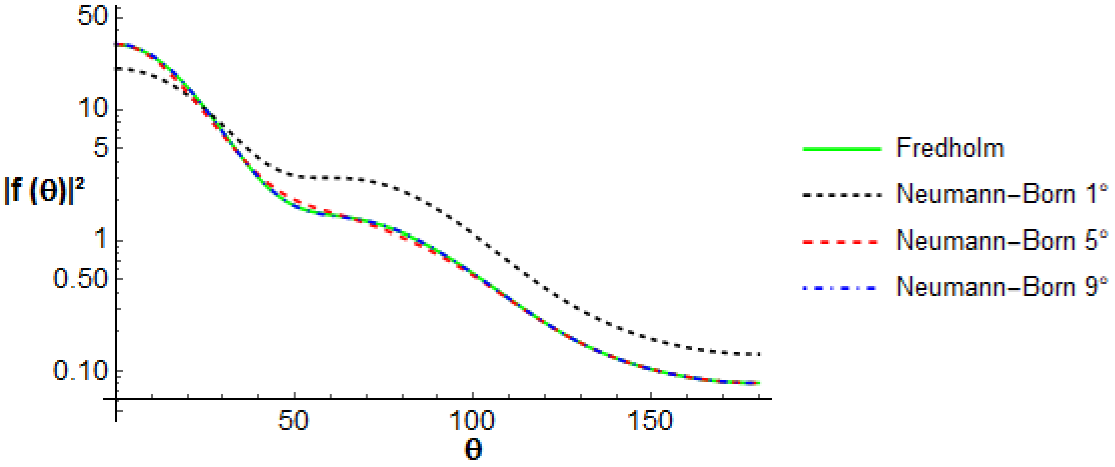


Figure 16- The Neumann-Born method approximates the Fredholm method.

Figure 17 is the graph of the absolute error of the arbitrary kernels and Neumann-Born method compared to the Born approximation on the Coulomb potential.

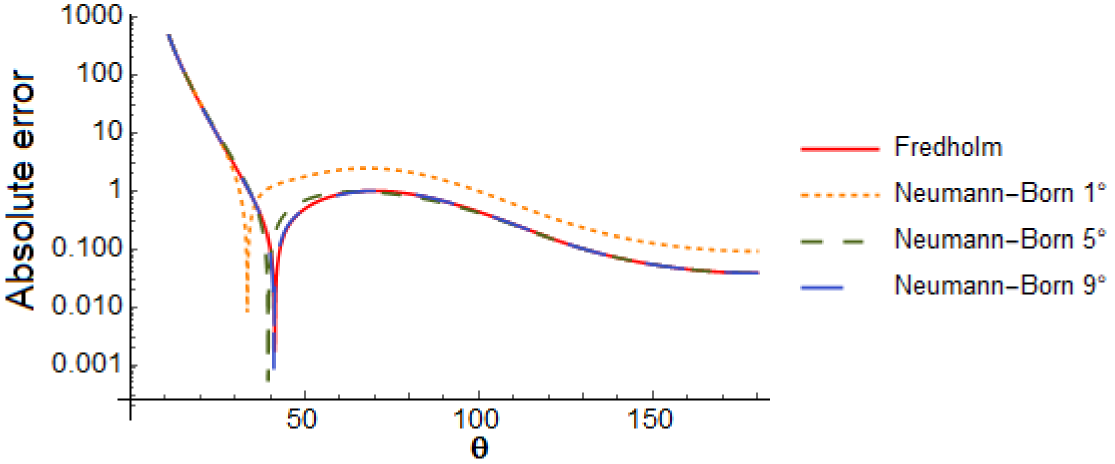


Figure 17- Absolute error between Fredholm and Neumann-Born methods.

It can also be seen in Figure 17 that the Neumann-Born method tends to the arbitrary kernel method as the series grade increases. In Figure 16, the closer to zero in the absolute error column, more the techniques approach the usual method, the Born approximation. For example, at the angle of 40, the error between the two techniques with respect to the Born approximation is 0.001.

Figure 18 shows the processing time for the arbitrary kernel method and the Neumann-Born method. For the arbitrary kernels method, size matrices were used 1×1 up to 5×5 (order matrix n) and for Neumann-Born, series 1 up to 5 order were generated, and for each case the processing time was calculated ten times to obtain its average.

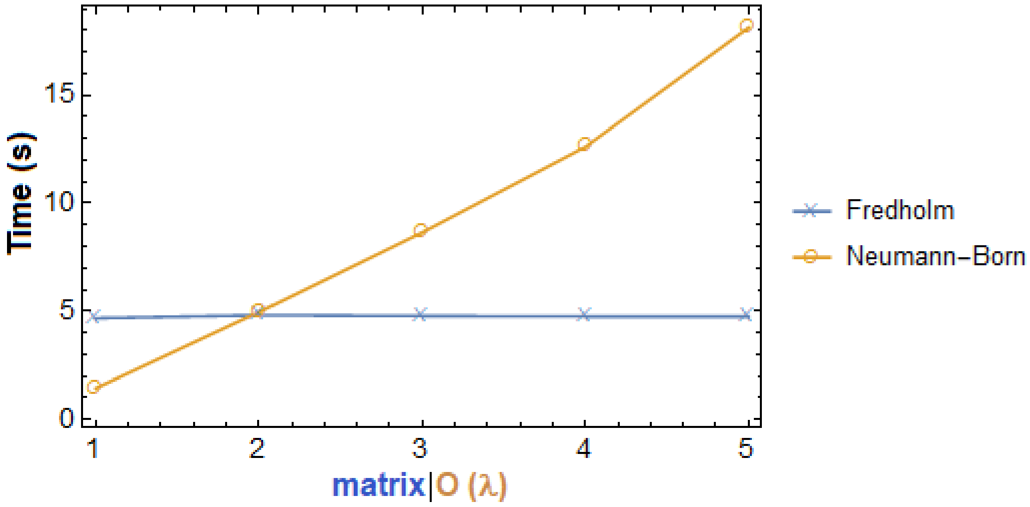


Figure 18- Processing time of the arbitrary kernel method and the Neumann-Born method.

In Figure 18, the Neumann-Born method shows linear growth and the arbitrary kernel method shows small variations. The arbitrary kernel method doesn’t show an increase in processing time because the determinant of matrices $n \times n$, $n \geq 2$, are null.

5. Conclusions

In this work we investigated the scattering of a particle at the Coulomb potential, where we use two codes to calculate the scattering amplitude of the Schrödinger integral equation, in addition to using a general hypothesis of the Born approximation. The results presented in Figure 13 and Figure 14 show us that the method generates a scattering with a behavior very similar to the Born approximation, but without the singularity close to the null angle and Figure 17 shows that the difference between the methods for the Born's approximation is between 0.1 and 1 for . $\theta \geq 40$.

As a future proposal, the analysis of the Fredhlom and Neumann-Born methods for the Yukawa potential (Yukawa, 1935), Podolsky (Podolsky, 1948) and the study of the Bethe-Salpeter integral equation in the light front (Sales, 2000).

Acknowledgements

JHS to CNPq 315519/2018-5, CAPES and FAPESB via PIE 0013/2016 for research Grant and PHS to PPGMC-UESC and CESUPA.

References

Arfken, G. B.; Weber, H. J. (2007). *Física matemática: Métodos matemáticos para engenharia e física*. Campus/Elsevier.

Bassalo, J.M.F, Cattani M.S.D (2012). *Elementos de física matemática*, 3 ed., Livraria da Física, São Paulo, Brasil.

Fredholm, I (1900). *Sur une nouvelle méthode pour la résolution du problème de Dirichlet*, Öfversight Kongl, Vetenskaps-Akad, Förhandlingar.

Geiger, H., Marsden, E. (1913). *The law deflexion of α particles throught large angles*, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science.

Griffiths, D.J. (2011). *Mecânica Quântica*, 2 ed., Pearson Education, São Paulo, Brasil.

Krasnov, M.I., Kisseliov, A.I., Makarenko, G.I. (1981). *Integral Equation*, Editora Mir, Moscow, Russian.

Nguyen, D.C. et. Al. (2011). *End-point energy measurements of field emisson current in a continuous-wave normal-conducting rf injector*, Physical Review Special Topics – Accelerators and Beams.

Girotto, P.S., Sales, J. H. (2019). *Fredholm Method for Podolsky Quantum Wave Function*, arXiv:1902.09990v1.

Podolsky, B., Schwed, P. (1948). *Review of a Generalized Eletrodynamics*, Rev. Mod. Phys 20,40.

Sakurai, J.J. (2013). *Modern Quatum Mechanics*, 2 ed., Addison-Wesley Publishing Company, California, USA.

Sales, J.H.O. et. al. (2000). *Light-front Bethe-Salpeter equation*, Physical Review. C. Nuclear Physics, 61 (2000) 044003.

Sales, J. H., Girotto, P.S., Soglia, L. V. M. (2021). *Computational Model of the Fredholm Method Applied to Podolsky Quantum Wave Function*. Brazilian Journal of Development, v.7, n.2, p.20386.

Smirnov, V. (1975). *Cours de Mathemátiques Supérieures. Tome IV*, Editions Mir, Moscow.

Yukawa, H. (1935). *On the interaction of elementary particles*. i. Proceedings of the Physico-Mathematical Society of Japan. 3rd series, v. 17, p. 48–57.