

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/333207381>

Symbolic and numerical study of Fourier series and PDEs using Maxima

Conference Paper · May 2019

CITATIONS

0

READS

173

2 authors:



[José Antonio Vallejo](#)

Universidad Politécnica de Cartagena

54 PUBLICATIONS 144 CITATIONS

[SEE PROFILE](#)



[Emmanuel Roque](#)

Center for Research and Advanced Studies of the National Polytechnic Institute

4 PUBLICATIONS 1 CITATION

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Control of Quantum Wave Equations [View project](#)



Relativistic heat equations [View project](#)

Symbolic and numerical study of Fourier series and PDEs using Maxima

José A. Vallejo¹, Emmanuel Roque¹

[emmanuelroquej@protonmail.ch]

¹ Faculty of Sciences, Autonomous University of San Luis Potosí, México

Fourier Analysis provide a set of techniques for solving partial differential equations(PDEs) in both bounded and unbounded domains, and various types of initial conditions. In the bounded domain case, the basic idea is to apply the separation of variables method which leads to a well-defined algorithm for developing the solution in a Fourier series. Therefore, this problem is tractable with a Computer Algebra System(CAS). In this work we introduce a Maxima package (called `pdefourier`) to solve it.

The package is able to compute the Fourier series of a function both numerically and symbolically, admitting piecewise-defined functions as arguments. It contains solvers for the one-dimensional heat and wave equations on a domain $[0, L]$, with general boundary conditions of the form

$$\begin{aligned}\alpha_1 u(0, t) + \beta_1 u_x(0, t) &= f_1(t) \\ \alpha_2 u(L, t) + \beta_2 u_x(L, t) &= f_2(t)\end{aligned}$$

Also, the package can solve the two-dimensional Laplace equation for a variety of domains (rectangles, disks, annuli, wedges) and boundary conditions (Dirichlet, Neumann and mixed).

Keywords: Fourier Analysis, PDEs, Mathematical Software.

1 Introduction

Let $f(x)$ be a piecewise-smooth function on the interval $[-L, L]$. Then, the Fourier coefficients of $f(x)$ are defined as follows:

$$\begin{aligned}a_n &= \frac{1}{L} \int_{-L}^L f(x) \cos\left(\frac{n\pi x}{L}\right) dx \\ b_n &= \frac{1}{L} \int_{-L}^L f(x) \sin\left(\frac{n\pi x}{L}\right) dx\end{aligned}$$

and the Fourier series of $f(x)$ is given by:

$$f(x) \sim \frac{1}{2}a_0 + \sum_{n=1}^{\infty} a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right)$$

Implementing routines for computing the Fourier coefficients in a CAS is not as straightforward as it might seem. Whenever the input function contains an expression involving sines and cosines whose arguments are integer multiples of the fundamental frequency or a product of polynomials with them, some of the coefficients must be computed separately as their general expression will have singularities for some integer value(s).

1 Example. If $f(x) = 3x^2 \cos(7x) + x$ on $[-\pi, \pi]$, using the trigonometric identity:

$$\cos \theta \cos \varphi = \frac{1}{2} \cos(\theta - \varphi) + \frac{1}{2} \cos(\theta + \varphi) \quad (1)$$

and integration by parts, a direct calculation shows that:

$$\begin{aligned} a_0 &= -\frac{12}{49} \\ a_n &= \begin{cases} \frac{12(n^2+49)(-1)^{n+1}}{(n-7)^2(n+7)^2} & n \neq 7 \\ \frac{98\pi^2+3}{98} & n = 7 \end{cases} \\ b_n &= \frac{2(-1)^{n+1}}{n} \end{aligned}$$

Another concern when dealing with trigonometric expressions is their multiple equivalences. This is a well-known issue in different CAS: due to the underlying algorithms in their symbolic integration routines, sometimes equivalent trigonometric expressions return different answers when calculating the Fourier coefficients. For instance, using equation 1 in the case $\theta = \varphi$ we get:

$$\cos(\theta)^2 = \frac{1}{2} + \frac{1}{2} \cos(2\theta)$$

For illustration purposes, these are the results given by Mathematica™ when you compute the Fourier cosine coefficients for each side of the equation.

```
In[1]:=FourierCosCoefficient[Cos[t]^2, t, n]
Out[1]:=0
In[2]:=FourierCosCoefficient[(1+Cos[2t])/2, t, n]
Out[2]:=1/2 (DiscreteDelta[-2 + n] + 2 DiscreteDelta[n])
```

The strategy followed in our implementation of the package is to transform any trigonometric function into its canonical form internally so that it becomes easy to decide whether or not the input contains an expression whose Fourier coefficients have singular values, using the pattern matching capabilities of the Maxima CAS, as described in the following section.

2 Computing the Fourier coefficients and the Fourier series

Because of the linearity of the integral and the use of trigonometric canonical forms, it is sufficient to detect the following patterns:

$$x^r \cos\left(\frac{m\pi x}{L}\right), x^r \sin\left(\frac{m\pi x}{L}\right), \cos\left(\frac{m\pi x}{L}\right), \sin\left(\frac{m\pi x}{L}\right), \quad r, m \in \mathbb{N}$$

To do so, we used the Maxima built-in commands `defmatch` and `matchdeclare`. Then, two similar strategies were followed depending on the input expression being piecewise-defined or not. The flow diagram appearing in Figure 1 shows the general idea behind both. Of course, when the input expression is piecewise-defined, the procedure described in the diagram must be followed in each interval of the domain.

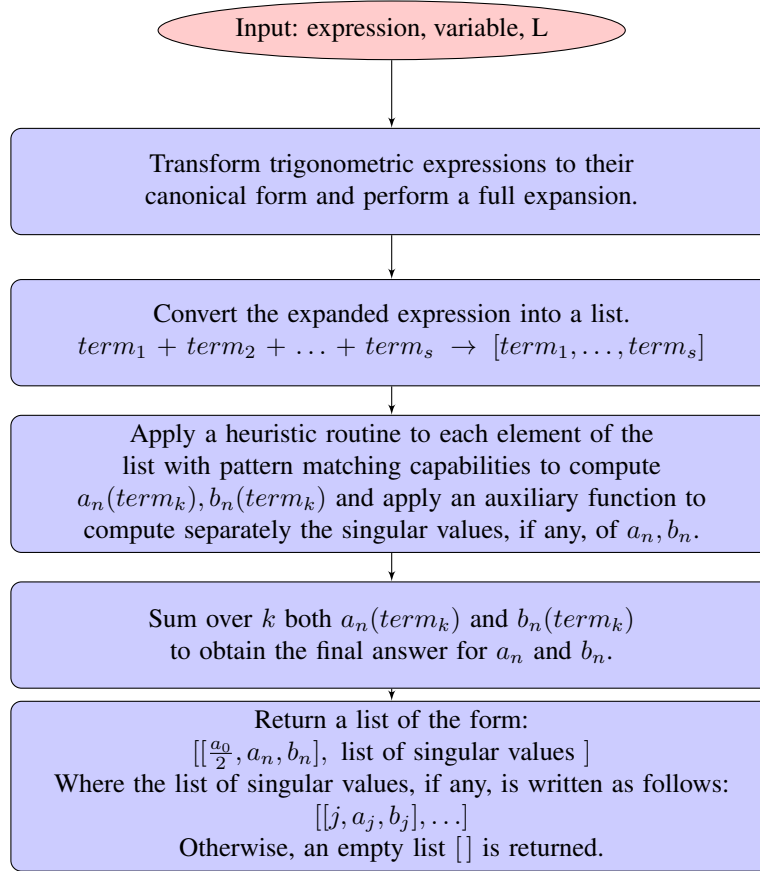


Figure 1: Routine for computing the Fourier coefficients.

The same ideas also apply for the case of the complex, sine and cosine coefficients, the only difference is in the way the answer is given (see Table 1). The Fourier series are obtained from an expansion routine of the Fourier coefficients. The upper limit of summation can be a positive integer or infinite. In the first case, a truncated series is returned; in the second, a symbolic series is displayed. We show some examples in the next section.

Type	Command	Answer format	List of s.v format
Trigonometric	fouriercoeff(expr, var, L)	$[[\frac{a_0}{2}, a_n, b_n], \text{list of s.v}]$	$[[j, a_j, b_j], \dots]$
Complex	cfouriercoeff(expr, var, L)	$[[c_0, c_n], \text{list of s.v}]$	$[[j, c_j], \dots]$
Cosine	fouriercoscoeff(expr, var, L)	$[[\frac{a_0}{2}, a_n], \text{list of s.v}]$	$[[j, a_j], \dots]$
Sine	fouriersincoeff(expr, var, L)	$[[b_n], \text{list of s.v}]$	$[[j, b_j], \dots]$

Table 1: Output formats for the Fourier coefficients (s.v= singular values).

2.1 Examples

Let $f(x)$ be as in Example 1. Then we have the following:

(% i2) `fouriercoeff(3*x^2*cos(7*x)+x,x,%pi);`

$$[[-\frac{6}{49}, \frac{12(n^2 + 49)}{n^4 - 98n^2 + 2401}, \frac{2(-1)^{n+1}}{n}], [[7, \frac{98\pi^2 + 3}{98}, \frac{2}{7}]]] \quad (\% \text{ o2})$$

Indeed, notice that $(n-7)^2(n+7)^2 = n^4 - 98n^2 + 2401$.

The case where MathematicaTM fails to compute correctly the cosine coefficients of equivalent trigonometric expressions, it is correctly solved by our package.

(% i3) `fouriercoscoeff(cos(x)^2,x,%pi);`

$$[[\frac{1}{2}, 0], [[2, \frac{1}{2}]]] \quad (\% \text{ o3})$$

(% i4) `fouriercoscoeff((1+cos(2*x))/2,x,%pi);`

$$[[\frac{1}{2}, 0], [[2, \frac{1}{2}]]] \quad (\% \text{ o4})$$

Here we have an example of a piecewise-defined function having singular values. We will use the command `cfouriercoeff` to get the complex Fourier coefficients, so that we can compare our answer to the one returned by MathematicaTM.

(% i5) `f(x):=if x>= -%pi and x<0 then x elseif x>=0 and x<%pi then sin(3*x)$`

(% i6) `cfouriercoeff(f(x),x,%pi);`

$$[[-\frac{3\pi^2 - 4}{12\pi}, \frac{(\%i\pi n^3 - 4n^2 - 9\%i\pi n + 9)(-1)^n - 2n^2 - 9}{2\pi n^2(n^2 - 9)}], [[3, -\frac{15\%i\pi - 4}{36\pi}]]] \quad (\% \text{ o6})$$

However, MathematicaTM is not able to detect the singular value of the coefficient when $n = 3$.

```
In[8]:=f[x]:= Piecewise[{{x,-Pi<x<0}, {Sin[3x],0<x<Pi}}]
In[9]:=FourierCoefficient[f[x], x, n]
```

$$\begin{cases} \frac{1}{3\pi} - \frac{\pi}{4} & n = 0 \\ \frac{(-1)^n (i\pi n^3 - 2((-1)^n + 2)n^2 - 9i\pi n - 9(-1)^n + 9)}{2\pi n^2(n^2 - 9)} & \text{True} \end{cases}$$

Now, let's see an example of how Fourier series are displayed symbolically. If $g(x) = x^4$ on $[-\pi, \pi]$ we get an answer like the ones appearing in Fourier Analysis textbooks.

(% i7) `fourier_series(x^4,x,%pi,inf);`

$$8 \left(\sum_{n=1}^{\infty} \frac{(\pi^2 n^2 - 6) (-1)^n \cos(nx)}{n^4} \right) + \frac{\pi^4}{5} \quad (\% \text{ o7})$$

Lastly, we show an example of how to use the expansion routines to obtain the Fourier series, truncated or not, and how they handle singular values in the Fourier coefficients when displaying an infinite series.

(% i8) `fcoeff:fouriersincoeff(x*cos(3*x),x,%pi);`

$$\left[\left[\frac{2n(-1)^n}{n^2 - 9} \right], \left[\left[3, -\frac{1}{6} \right] \right] \right] \quad (\text{fcoeff})$$

(% i9) `fouriersincoeff_expand(fcoeff,x,%pi,5);`

$$-\frac{5 \sin(5x)}{8} + \frac{8 \sin(4x)}{7} - \frac{\sin(3x)}{6} - \frac{4 \sin(2x)}{5} + \frac{\sin(x)}{4} \quad (\% \text{ o9})$$

(% i10) `fouriersincoeff_expand(fcoeff,x,%pi,inf);`

The sum is over \mathbb{N} - \{3\}

$$2 \left(\sum_{n=1}^{\infty} \frac{n(-1)^n \sin(nx)}{n^2 - 9} \right) - \frac{\sin(3x)}{6} \quad (\% \text{ o10})$$

It is important to notice that displaying infinite series correctly has been a source of troubles in different CAS. This is mainly due to the fact that they are not evaluated, only displayed symbolically, and so, a very human simplification might not be performed by a CAS. For illustration purposes, suppose that we want to compute the Fourier series of $h(x) = \sin 15x$ on the interval $[-\pi, \pi]$. It is obvious that its Fourier series is exactly equal to $h(x)$, because its Fourier coefficients are $a_0 = 0$, $a_n = 0$, $b_n = \delta_n^{15}$ and we have:

$$h(x) \sim \sum_{n=1}^{\infty} \delta_n^{15} \sin 15x = \sin 15x$$

However, infinite sums of expressions containing Kronecker delta functions are really hard to simplify to a single term, because it must be verified that the only index that does not vanish is indeed contained in the set of indices over which you are considering the sum. Integration routines in different CAS sometimes return a result for b_n in terms of some sort of Kronecker

delta (Mathematica™ returns it in terms of `DiscreteDelta`), which complicates the task of obtaining the Fourier series by an expansion of the coefficients.

Following our approach, we avoid the issue of the evaluation of Kronecker delta functions inside an infinite sum:

```
(% i10) fourier_series(sin(15*x),x,%pi,inf);
                                sin(15x)                                (% o10)
```

However, as we just saw in the examples above, when the list of singular values is not empty, we have to print a message indicating the set of indices that are taken into account in the infinite sum.

3 Partial differential equations

In this work, we focused on solving the three main second-order linear partial differential equations with constant coefficients, namely, the heat, Laplace's and wave equations. The mathematical details can be found in many standard textbooks on PDEs or Fourier Analysis. Here, we will discuss some of the technicalities of the implementation and a list of examples.

Heat equation	$u_t(x, t) = \kappa u_{xx}(x, t) + Q(x, t)$
Laplace's equation	$u_{tt} + u_{xx} = 0$
Wave equation	$u_{tt}(x, t) = c^2 u_{xx}(x, t) + F(x, t)$

Table 2: PDEs solved by `pdefourier`

In the previous section, it was mentioned that Fourier series are obtained by performing an expansion of the Fourier coefficients. This approach is multipurpose. We've already seen that it allows a user to expand a list of Fourier coefficients that have been computed previously, but it is also useful to code the solution of the PDEs in an easier way. Although the package can solve the heat equation with a heat source $Q(x, t)$, we want to show how the expansion routines facilitates the process of finding solutions with an easier version of the equation. Consider the following IBVP:

$$\begin{cases} u_t = \kappa u_{xx} & (x, t) \in [0, L] \times \mathbb{R}^+ \\ u(0, t) = 0 & u(L, t) = 0 \\ u(x, 0) = f(x) \end{cases}$$

Then, the solution is given by

$$u(x, t) = \sum_{n=1}^{\infty} B_n \exp[-n^2 (\frac{\pi \kappa}{L})^2 t] \sin \frac{n\pi x}{L}$$

$$B_n = \frac{2}{L} \int_0^L f(x) \sin \frac{n\pi x}{L}$$

Notice that for our purposes, it will be sufficient if we create a list of the form $[[C_n(t)], [j, C_j(t)]]$ where $C_n(t) = B_n \exp[-n^2 (\frac{\pi \kappa}{L})^2 t]$ and then use the expansion routine corresponding to a

Fourier sine series on the interval $[0, L]$ and the space variable x . For instance, if we consider $f(x) = x^2(1 - x)$, $L = 1$, $\kappa = 1$, a solution to the IBVP can be obtained in a few lines of code:

```
(% i11) f(x):=if (0<=x and x<=1) then x^2*(1-x)$
(% i12) [[Bn],lsv]:fouriersincoeff(f(x),x,L)$
(% i13) Cn:Bn*exp(-n^2*pi^2*t)$
(% i14) fouriersincoeff_expand([[Cn],[ ]],x,L,inf);
```

$$-\frac{4 \sum_{n=1}^{\infty} \frac{(2(-1)^n + 1) e^{-\pi^2 n^2 t} \sin(\pi n x)}{n^3}}{\pi^3} \quad (\% \text{ o14})$$

Here the list of singular values was empty (obviously), and no more work was required. Similar methods to obtain solutions for the three equations with different types of boundary conditions and domains were implemented, dealing with general expressions (piecewise defined or not) and taking care of the possible singular values in the coefficients.

3.1 Examples

We will show one example for each equation, in all of them, Mathematica™ is not able to find the solution. Although `pdefourier` is able to solve these equations with many more boundary conditions, the ones appearing next are meant to give a general idea about how to use the package to solve PDEs.

Heat equation: Consider the Dirichlet problem:

$$\begin{cases} u_t = \kappa u_{xx} + Q(x, t) & (x, t) \in [0, L] \times \mathbb{R}^+ \\ u(0, t) = h_1(t) \quad u(L, t) = h_2(t) \\ u(x, 0) = F(x) \end{cases}$$

The syntax for solving it using `pdefourier` is

`dirichlet_heat(Q(x,t),F(x),h1(t),h2(t),x,t,%pi,%kappa,ord).`

If $h_1(t) = A$, $h_2(t) = B$, $A, B \in \mathbb{R}$ and $Q(x, t) = \kappa$ the solution is given by:

```
(% i15) declare(F,real)$
(% i16) declare(L,constant)$
(% i17) assume(L>0)$
(% i18) dirichlet_heat(%kappa,F(x),A,B,x,t,L,%kappa,inf);
```

$$\sum_{n=1}^{\infty} \frac{e^{-\frac{\pi^2 n^2 t}{L^2}} \sin\left(\frac{\pi n x}{L}\right) \left(2 \pi^3 n^3 \int_0^L F(x) \sin\left(\frac{\pi n x}{L}\right) dx + (2L^3 - 2L^3(-1)^n) e^{-\frac{\pi^2 n^2 t}{L^2}} + (2\pi^2 L B n^2 + 2L^3(-1)^n - 2\pi^2 L A n^2 - 2L^3) \right)}{\pi^3 L} - \frac{(A-B)x}{L} + A$$

Laplace's equation: Here we consider the Laplace's equation on a wedge and Neumann conditions:

$$\begin{cases} u_{rr} + \frac{1}{r}u_r + \frac{1}{r^2}u_{\theta\theta} = 0 & (r, \theta) \in [0, R] \times [0, \alpha], 0 < \alpha < 2\pi \\ u(r, 0) = 0 & u(r, \alpha) = 0 \\ u_r(R, \theta) = f(\theta) \end{cases}$$

The syntax for solving it using `pdefourier` is `neumann_laplace_wedge(R, %alpha, f(theta), theta, ord)`. In this example we will consider $\alpha = \pi/2$ and $R = 1$.

```
(% i19) declare(f,real)$
(% i19) neumann_laplace_wedge(1,%pi/2,f(%theta),%theta,inf);
```

$$\frac{2 \sum_{n=1}^{\infty} \frac{\sin(2\theta n) \int_0^{\frac{\pi}{2}} f(\theta) \sin(2\theta n) d\theta r^{2n}}{n}}{\pi} \quad (\% \text{ o19})$$

Wave equation: For this example, we will consider the wave equation with homogeneous boundary conditions and a driving term.

$$\begin{cases} u_{tt}(x, t) = c^2 u_{xx}(x, t) + F(x, t) & (x, t) \in [0, L] \times \mathbb{R}^+ \\ u(0, t) = 0 & u(L, t) = 0 \\ u(x, 0) = f(x) & u_t(x, 0) = g(x) \end{cases}$$

The syntax for solving it using `pdefourier` is `fourier_wave(F(x, t), f(x), g(x), x, t, L, c, ord)`. If $F(x, t) = \cos(\omega t)r(x)$ and $g(x) = 0$, then:

```
(% i20) assume(t>0)$
(% i21) declare(f,real)$
(% i22) declare(r,real)$
(% i23) fourier_wave(r(x)*cos(%omega*t),f(x),0,x,t,L,c,inf);
```

$$-\sum_{n=1}^{\infty} \frac{\sin\left(\frac{\pi n x}{L}\right) \left(\left(2 L^2 \cos\left(\frac{\pi c n t}{L}\right) - 2 L^2 \cos(\omega t) \right) \int_0^L r(x) \sin\left(\frac{\pi n x}{L}\right) dx + (2 L^2 \omega^2 - 2 \pi^2 c^2 n^2) \cos\left(\frac{\pi c n t}{L}\right) \int_0^L f(x) \sin\left(\frac{\pi n x}{L}\right) dx \right)}{\pi^2 L c^2 n^2 - L^3 \omega^2}$$

Since no assumptions were made about ω , the solution corresponds to the case without resonance ($\omega \neq \frac{cn\pi}{L}$).

4 Conclusions

In this work we have discussed the main challenges of implementing a program that can compute Fourier series correctly and the strategies followed to tackle them. We also showed

how such program became the basis to a larger set of functions that solve the three main second-order partial differential equations with different types of boundary conditions and its comparable performance to commercial software. The election of the free software Maxima to develop it is highly important because the availability of the code can play a key role in obtaining feedback from the mathematical community about the efficiency and correctness of the algorithms, and its functionality can benefit from users' contributions in the future to extent the program capabilities and solve symbolically more partial differential equations, in higher dimensions and in a wider variety of domains.

References

- [1] JAMES S. WALKER, *Fourier Analysis*. Oxford University Press, New York, 1988.
- [2] R. HABERMAN, *Applied Partial Differential Equations*. Pearson Education, Upper Saddle River (NJ), 2013.
- [3] MAXIMA.SOURCEFORGE.NET, Maxima, a Computer Algebra System, Version 5.42 (2019). <http://maxima.sourceforge.net/>