A Numerically Stable Fourier Continuation Approximation for the Solution of Partial Differential Equations

Melanie Vining

Integrated Applied Mathematics
Department of Mathematics & Statistics
University of New Hampshire

March 22, 2019

Overview

Motivation

The combining of an Alternating Direction approach with Fourier Continuation approximations (FC-AD) splits a PDE into a series of one-dimensional BVPs which are then solved using Fourier Continuation approximations ([1], [2]). This research develops and shows a new approach to Fourier Continuation approximation that offers greater stability and accuracy.

FC-AD Overview

- Based on Alternating Direction Implicit (ADI) methods
- Splits the PDE into a series of (the same) BVP
- Results in stability as long as the solution to the BVP is stable under that operation

FC-AD Computation

Given the 2-D Heat Equation

$$\begin{aligned} u_t = k(u_{xx} + u_{yy}) + Q(x,y,t), &\quad (x,y,t) \in \Omega \times (0,T], \\ u(x,y,t) = G(x,y,t) &\quad (x,y) \in \partial\Omega, t \in (0,T] \\ u(x,y,0) = u_0(x,y), &\quad (x,y) \in \Omega, \end{aligned} \tag{1.1}$$

FC-AD

- Discretize in time, with $t^n = n\Delta t$ and use central difference centered on $t^{n+\frac{1}{2}} = (n+\frac{1}{2})\Delta t$.

$$\frac{u^{n+1}-u^n}{\Delta t} = \frac{k}{2}\frac{\partial^2}{\partial x^2}(u^{n+1}+u^n) + \frac{k}{2}\frac{\partial^2}{\partial y^2}(u^{n+1}+u^n) + Q^{n+\frac{1}{2}} + E_1(x,y,\Delta t) \tag{1.2}$$

- Solve for u^{n+1}:

$$\left(1 - \frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2} - \frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)u^{n+1} = \left(1 + \frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2} + \frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)u^n + Q^{n+\frac{1}{2}} + E_1(x,y,\Delta t) \tag{1.3}$$

FC-AD

Expanding and arranging terms gives

$$\left(1-\frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\left(1-\frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)u^{n+1}$$
$$=\left(1+\frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)\left(1+\frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)u^{n}+\frac{k^2\Delta t^2}{4}\frac{\partial^2}{\partial x^2}\frac{\partial^2}{\partial y^2}(u^{n+1}-u^{n})$$
$$+\Delta t Q^{n+\frac{1}{2}}+\Delta t E_1(x,y,\Delta t) \tag{1.4}$$

Here, inverting the operators $\left(1-\frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2}\right)$ and $\left(1-\frac{k\Delta t}{2}\frac{\partial^2}{\partial y^2}\right)$ gives an expression for u^{n+1} in terms of some f^{n+1}.

FC-AD

This results in

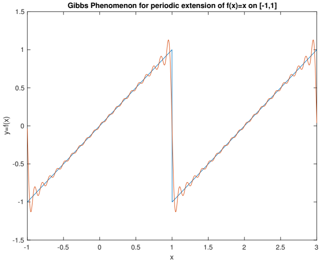
$$\left(1 - \frac{k\Delta t}{2}\frac{\partial^2}{\partial x^2}\right) u = f \tag{1.5}$$

Inverting this operator results in the Boundary Value Problem

$$-\alpha u'' + u = f, \quad u(x_\ell) = B_\ell, \quad u(x_r) = B_r \tag{1.6}$$

Motivation for using FC

- Fourier Series are known to represent periodic functions well, with cheap computational cost
- Fourier Series used to represent non-periodic functions experience Gibbs' Phenomenon



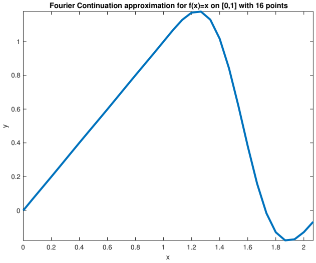
- Goal of Fourier Continuation is to get the similar accuracy, convergence, and computational cost with non-periodic functions
- As a result of using Fourier Continuation approximations, we are given periodic extensions of smooth functions that can be used in conjunction with FFTs to compute numerical solutions to differential equations.

Construction

Given a smooth function $y = f(x)$ on $[0, 1]$ where $f \in C^k[0, 1]$, $k > 0$. Discretize the interval onto N points and let x_j be the j^{th} point on the interval with corresponding function value $y_j = f(x_j)$. Choose period $b > 1$, and expand in terms of M Fourier Modes, $M < N$, and the problem to solve becomes

$$y_j = \sum_{k \in t(M)} a_k e^{\frac{2\pi i}{b} k x_j}, \quad j = 1 \dots N \tag{1.7}$$

[3]. As an example, consider $f(x) = x$ on $[0, 1]$. We discretize on 16 points, let $b = 2$, and $M = 7$. A resulting Fourier Continuation approximation for f is



FC-Gram

- Begin with a basis of polynomials on n points, $f_0, f_1, \ldots, f_{n-1}$
- Create and store the Fourier Continuation approximations for each of the polynomials
- Given a function f on any interval with any number of points m, sample the first n points and the last n points of f. If $m = n$, use the same n points in order.
- Fit the first n points to the basis and the last n points to the basis separately, giving us coefficient vectors a_1 and a_2 respectively.
- An appropriate combination of a_1 and a_2 applied to the continuations of their respective polynomials is taken to get a smooth, periodic extension of f

FC-Gram Example

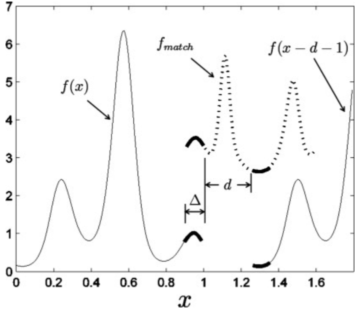
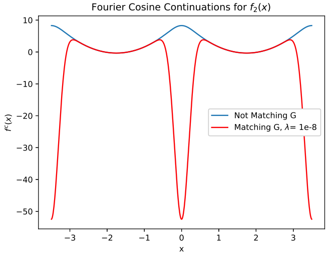


Figure: A comprehensive picture of the FC-Gram method as applied to $f(x) = e^{\sin(5.4\pi x - 2.7\pi) - \cos(2\pi x)}$[1]

Overview

Multiple Continuation approximations

As explained in the Fourier Continuation section, there are several possible Fourier Continuation approximations for the same function.



By adding constraints to the system of equations, we can control the shape of the continuation and ultimately create a stable approximation.

Green's Function Solutions

Matching the differential operator to the Green's Function solutions for the given BVP is a natural choice. The Green's Function solutions are inherently stable under the ∞-norm and thus provide the beginning steps for the new form the continuations will take.

- Compute the Green's Function for the differential equation $-\alpha u'' + u = f$, $u(x_\ell) = 0$, $u(x_r) = 0$. This yields $G(x, a)$ [4].
- Compute the Green's Function solution as $\int_{x_\ell}^{x_r} G(x, a) f(a) da$.

The homogeneous solutions used to calculate the Green's function for the given BVP with $x_\ell = -1$, $x_r = b$ are

$$\begin{aligned} h_1(x) &= e^{\frac{(x-b)}{\sqrt{\alpha}}} && (2.1) \\ h_2(x) &= e^{\frac{(-x-1)}{\sqrt{\alpha}}}, && (2.2) \end{aligned}$$

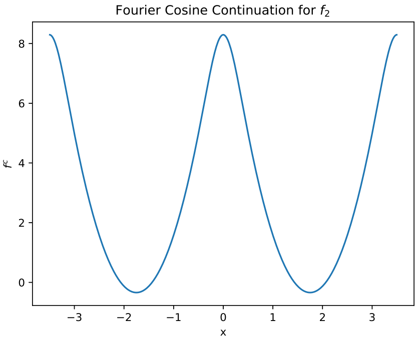
The calculated Green's Function for $x_\ell = -1$, $x_r = b$ is

$$G(x,a) = \begin{cases} -\frac{1}{2}\dfrac{\sqrt{\alpha}\left(-e^{\frac{-1-2b+a}{\sqrt{\alpha}}} + e^{-\frac{a+1}{\sqrt{\alpha}}}\right)\left(-e^{\frac{2(2+b+x)}{\sqrt{\alpha}}} + e^{\frac{2(x+1)}{\sqrt{\alpha}}} + e^{\frac{2(1+b)}{\sqrt{\alpha}}} - 1\right)e^{\frac{-x+1+2b}{\sqrt{\alpha}}}}{\left(e^{\frac{2(1+b)}{\sqrt{\alpha}}} - 1\right)^2} & x < a \\ \frac{1}{2}\dfrac{\sqrt{\alpha}\left(e^{\frac{2+b+a}{\sqrt{\alpha}}} - e^{\frac{a-b}{\sqrt{\alpha}}}\right)\left(e^{\frac{2(-x+b)}{\sqrt{\alpha}}} + e^{\frac{2(1+b)}{\sqrt{\alpha}}} - e^{\frac{2+4b-2x}{\sqrt{\alpha}}} - 1\right)e^{\frac{2+b+x}{\sqrt{\alpha}}}}{\left(e^{\frac{2(1+p)}{\sqrt{\alpha}}} - 1\right)^2} & x \geq a \end{cases} \quad (2.3)$$

- Given n points on an interval, we construct the Gram polynomials $f_0, f_1, \ldots, f_{n-1}$. Then a sine or cosine continuation matrix is created on a grid, and the functions are evaluated on the same grid.
- Next, the function $\tilde{f}$ and its cosine (or sine) continuation C_u are matched, which is the equivalent of solving

$$\min ||C_u f^c - \tilde{f}||^2. \tag{2.4}$$

This results in an accurate, but generally unstable approximation:

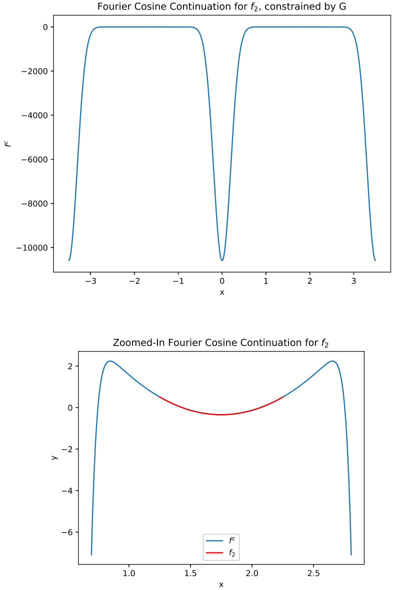


Minimization Problem 2

- Using symbolic representations, the Green's function solutions for each of the polynomials are computed and evaluated on the grid. Then a numerical differential operator $I - \alpha \frac{\partial^2}{\partial x^2}$ is created for the sine or cosine continuation.
- We constrain the original minimization problem using the Green's function and the inverted differential operator applied to the continuation. This yields

$$\min ||C_u f^c - \tilde{f}||^2 + ||C_u D_c^\dagger f^c - \tilde{G}||^2 \tag{2.5}$$

While this approximation is stable, it loses accuracy.



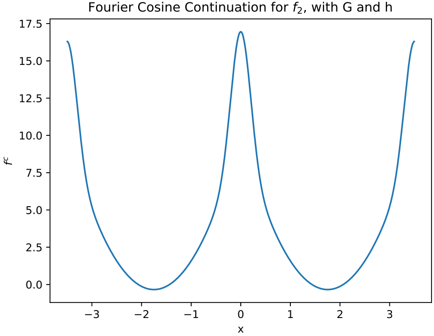
Minimization Problem 3

In practice, solving this minimization problem does not achieve great accuracy nor stability. The homogeneous solutions are not well-represented in the Fourier basis, but the Green's function solution is created from them. An additional constraint is needed in order to allow the system to accurately represent the function while separating out the homogeneous solutions. Including this constraint modifies the system to

$$\min ||C_u f^c - \tilde{f}||^2 + ||C_u D_c^{\dagger} f^c + c_1 \tilde{h_1} + c_2 \tilde{h_2} - \tilde{G}||^2 \tag{2.6}$$

where c_1 and c_2 will be chosen by the computational method to achieve greatest accuracy in the least-squares sense.

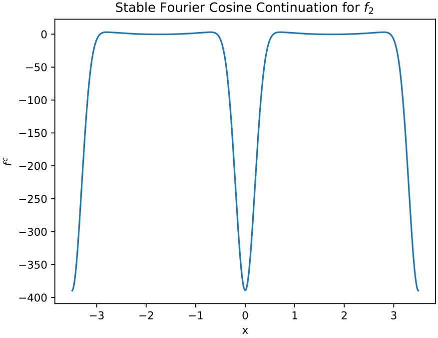
This gives accuracy, but trades off on stability.



Minimization Problem 4

Lastly, two other parameters λ and μ which force the continuations f^c to be stable. The first, λ, is applied to control the magnitude of c_1 and c_2 relative to the homogeneous solutions. The second, μ is applied to the coarse-grid point values of the differential operator to control the magnitude of the continuation under the derivative. The final minimization problem we have is

$$\begin{aligned} \min ||C_u f^c - \tilde{f}||^2 + ||C_u D_c^\dagger f^c + c_1 \tilde{h_1} + c_2 \tilde{h_2} - \tilde{G}||^2 \\ +\lambda^2 ||c_1 \tilde{h_1} + c_2 \tilde{h_2}||^2 + \mu^2 ||C_u D_c^\dagger|_{\text{coarse}}||^2 \end{aligned} \tag{2.7}$$


Stable Fourier Cosine Continuation for f_2
0
−50
−100
−150
−200
−250
−300
−350
−400
f^c
−3
−2
−1
0
1
2
3
x

Implementation

We implement this solver in Julia, using 200 bits of precision in `BigFloat`. For 200 bits, $\epsilon \approx 10^{-77}$. With the conditioning of the system, having 77 digits of precision mitigates great error propagation. The Julia packages `GenericSVD`, `SymPy`, and `PyPlot` were used. To find the least-squares solution to our minimization problem, we used an SVD and found the pseudoinverse with a tolerance of 10^{-40}.

Implementation 2

Let n be the number of points on a coarse grid unit interval, coinciding with the number of points the Gram polynomials are computed on. Let F be the factor by which the coarse grid is refined, so if the original spacing is $h = \frac{1}{n-1}$, the fine grid spacing will be $h_f = \frac{1}{(n-1)\times F}$. While calculations are performed on a full double period $(2b)$, measurable results are only calculated on the unit interval.

Accuracy was measured as

$$a = \left\| f^c\big|_{\text{fine}} - f\big|_{\text{fine}} \right\|_2 \tag{2.8}$$

and stability was measured as

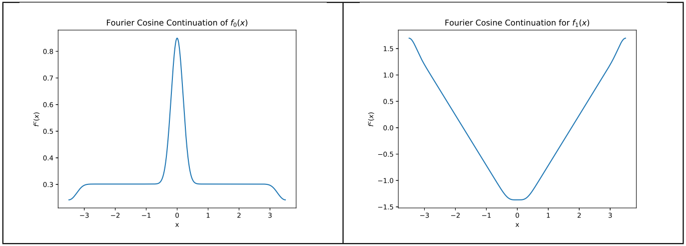
$$s = \frac{\left\| u^c\big|_{\text{coarse}} \right\|_2}{\left\| f\big|_{\text{coarse}} \right\|_2} \tag{2.9}$$

Some Results

Letting $n = 10$, $b = 3.5$ and setting the unit interval from 1.25 to 2.25 the evaluation of the Gram polynomials and the corresponding Green's function solutions were shifted accordingly . The coarse grid of 10 points was refined by a factor of 10 to give a fine grid of 100 points. Stability was achieved with great accuracy for various values of α.

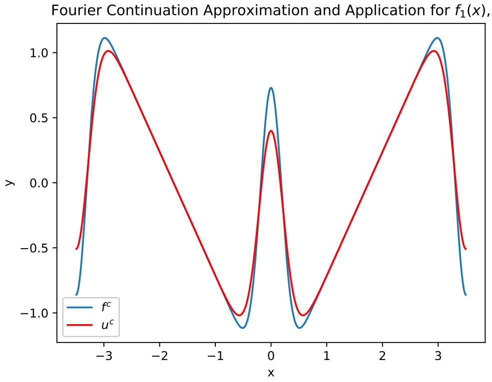
Some Results

Here are two cosine continuations for f_0 and f_1 with $\alpha = 0.001$:



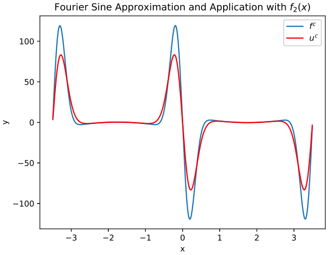
Some Results

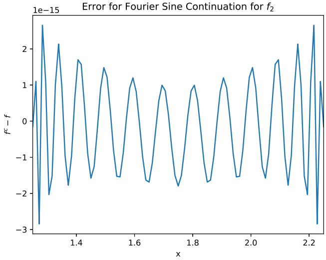
Here is a cosine continuation for f_1, accompanied by the computed solution u, for $\alpha = 0.01$



Some Results

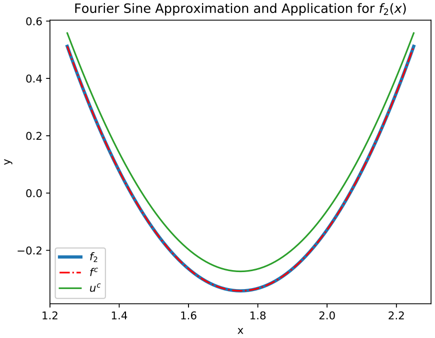
Here is a sine continuation for f_2, accompanied by the error for the unit interval a for $\alpha = 0.01$ with $\lambda = 3.59 \times 10^{-14}$. This is a stable approximation.





Some Results

To better view how the sine continuation matches the function, here is a zoomed in view of f_2, f_2^c and the accompanying solution u for $\alpha = 0.01$ with $\lambda = 3.59 \times 10^{-14}$, on the unit interval f_2 is originally defined on:



Predicting λ and μ given α

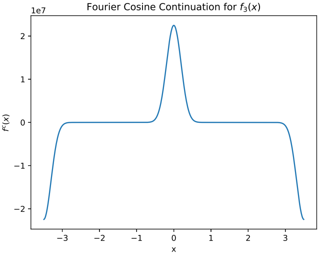
The stability of the approximations is dependent on α. To be able to use this method as a fast method, the form of the continuation must be easily predicted for any given α. The first approach taken was to find a value of λ that gives stability up to a tolerance. This was achieve with a bisection method for values of λ between 10^{-20} and 10^{0}.

For Fourier Cosine continuations with $\alpha = 0.001$, the optimized λ values are listed with the corresponding measure of accuracy and stability for each polynomial:

Degree	λ	Accuracy	Stability
0	$3.255881466810751 \times 10^{-13}$	$2.86510896623681 \times 10^{-17}$	0.999999999999
1	$1.839023251669854 \times 10^{-13}$	$3.92206770433105 \times 10^{-17}$	0.999999999999
2	$1.4852300108136326 \times 10^{-10}$	$3.88957362362112 \times 10^{-12}$	0.999999999999
3	$5.518469567738165 \times 10^{-10}$	$5.61719253880104 \times 10^{-11}$	0.999999999999
4	$1.1767778737499883 \times 10^{-9}$	$2.0963541751156 \times 10^{-10}$	0.999999999999
5	$3.0091447070493457 \times 10^{-9}$	$1.4098065193636 \times 10^{-9}$	0.999999999999
6	$4.864688165149565 \times 10^{-9}$	$3.1835304451683 \times 10^{-9}$	0.999999999999
7	$1.0373945031909215 \times 10^{-8}$	$1.52729276442667 \times 10^{-8}$	0.999999999999
8	$1.5639865553913073 \times 10^{-8}$	$3.08554433657289 \times 10^{-8}$	0.999999999999
9	$3.071964692890429 \times 10^{-8}$	$1.72857119028618 \times 10^{-7}$	0.999999998446

Controlling the Continuation Size

As a result of maximizing stability, the magnitude of the continuations becomes large.



This is not ideal for implementation in other programming languages as the magnitude will affect the number of digits of accuracy that can be maintained when using a language that has $\epsilon < 10^{-77}$.

Overview

- Now that stability is an achievable result, we hope to find a predictor such that a continuation can be stored and chosen based on the value of α used, without solving a system every time
- After the basis is stored, the FC-Gram method will be applied to several test functions, including the Runge function, in order to determine the accuracy of the method in implementation with any input of data
    - At this point, the FC-Gram method will be applied first on n points, and then increasing the number of points to get an error estimate as the number of points gets large.
- Next, BVPs with various BCs with be solved using the continuations and FC-Gram.
- Lastly, this will be implemented in PDEs as the BVP-solver in the time stepping of the FC-AD method

- A convergence estimate is needed first and foremost. It has been shown in [5] that convergence of fast stable approximations on equispaced points is impossible, therefore the convergence estimates speaks to maximal errors as step size decreases instead of taking a limit.
- Formal justification of the stability of this method is needed.

Overview

References

Oscar P. Bruno and Mark Lyon.
High-order unconditionally stable fc-ad solvers for general smooth domains i. basic elements.
Journal of Computational Physics, 229:2009–2033, 2010.

Mark Lyon and Oscar P. Bruno.
High-order unconditionally stable fc-ad solvers for general smooth domains ii. elliptic, parabolic and hyperbolic pdes; theoretical considerations.
Journal of Computational Physics, 229:3358–3381, 2010.

Oscar P. Bruno, Youngae Han, and Matthew M. Pohlman.
Accurate, high-order representation of complex three-dimensional surfaces via fourier continuation analysis.
Journal of Computational Physics, 227:1094–1125, 2007.

Carl M. Bender and Steven A. Orszag.
Advanced Mathematical Methods for Scientists and Engineers; Asymptotic Methods and Perturbation Theory.
Springer, 1999.

Rodrigo B. Platte, Lloyd N. Trefethen, and Arno B.J. Kuijlaars.
Impossibility of fast stable approximation of analytic functions from equispaced samples.
SIAM Review, 53(2):308–318, 2011.