Harmonic Balance for Differential Equations with Periodic Solutions

Aubrey Jaffer

December 2005

Abstract

The Fourier transform of a periodic function has only a countable number of non-zero coefficients. Thus, casting a system of differential equations in a periodic space radically reduces the number of possible solution functions. While Fourier series solve linear differential equations with constant coefficients, a method hinted by the *Harmonic Balance* analysis of nonlinear electrical circuits extends the solution space to periodic square-integrable functions.

This framework employs a succinct representation for approximating square-integrable periodic functions. These function objects can be scaled, added, differentiated, and integrated in time linear in the number of their stored coefficients; multiplication of two function objects can be accomplished in sub-quadratic time using matrix multiplication techniques.

Development

Linear electrical networks are analyzed by applying the Laplace transform to the node and mesh equations; algebraically or iteratively solving the resulting system of equations; then applying the inverse Laplace transform to that solution. In the iterative case, only one frequency is analyzed at a time; each node and branch has a complex voltage or current for that frequency.

In *harmonic balance* analysis, associated with each node are one or more pairs of frequency and voltage. For each frequency, linear components process voltages as before; assuming a voltage of 0 for frequencies not present in a node's list. The basic non-linear operator is multiplication (*mixing*), which produces coefficients for sum and difference frequencies according to sum-of-angles trigonometric identities.

Multiple products can produce positive integer powers. An arbitrary differentiable function can be synthesized as a Taylor or power series.

For Hamiltonian systems, the energy lost replacing small coefficients with zero can be distributed among the nonzero coefficients (to preserve the energy invariance).

The sum of squared-magnitude differences between corresponding elements of two iterations of a Fourier coefficient list is a reasonable metric for gaging convergence.

Representation

A function $f(\vec{x})$ is represented by its nonzero Fourier series coefficients stored as a list of pairs, each linking an exact vector frequency, $\vec{w} = [w_1, w_2, \ldots, w_n]$, with an inexact complex coefficient, $F[\vec{w}]$.

Calculated coefficients having magnitudes less than some threshold (which may change during the computation) are treated as zeros and not stored. Because the Fourier transform is an orthogonal transform, the error introduced by zeroing a coefficient is proportional to its magnitude squared. Zeroing based on the magnitude of a coefficient introduces less error per stored coefficient than series truncation at some predetermined point.

This representation is not dependent on a grid. A function object denotes that function over its whole (modular) range.

Operations

The software procedures to scale, add, differentiate, integrate, and multiply take coefficient lists as arguments and return a coefficient list.

Because the Fourier Transform is linear,

$$s(\vec{x}) = \alpha f(\vec{x}) + \beta g(\vec{x}) \Longleftrightarrow S(\vec{w}) = \alpha F[\vec{w}] + \beta G[\vec{w}]$$

and

$$r(\vec{x}) = \frac{\partial^k f(\vec{x})}{\partial {x_j}^k} \Longleftrightarrow R(\vec{w}) = (2\pi i w_j)^k F[\vec{w}]$$

and

$$q(\vec{x}) = \int^k f(\vec{x}) d{x_j}^k \Longleftrightarrow Q(\vec{w}) = \frac{F[\vec{w}]}{(2\pi i w_j)^k}$$

where w_j is the jth component of $\vec{w}$. In integration, vector frequencies with components $w_j = 0$ must receive different treatment.

A function is a linear combination of sine waves; so products of functions behave thus:

$$p(\vec{x}) = f(\vec{x})g(\vec{x}) \Longleftrightarrow P(\vec{w}) = \frac{1}{2} \sum_{\vec{v}=\vec{0}}^{\vec{w}} F[\vec{v}]\, G[\vec{w} - \vec{v}] + F[\vec{v}]\, G[\vec{w} + \vec{v}]$$

where $\vec{v}$ ranges over vectors with components $0 \leq \vec{v}_j \leq \vec{w}_j$. $G[u_0, u_1, \ldots u_n]$, where some of the u_j are negative, is $-1^m G[|u_0|, |u_1|, \ldots |u_n|]$, where m is the number of negative u_j.

We now have all the operations needed to solve or simulate differential equations entirely in the Fourier dual-space. Non-polynomial functions can be approximated using Taylor or power series.

Solving

The proposed framework should be well suited to field calculations for crystalline materials with rectangular lattices. Other crystal lattices can be used with larger spatial periods; better would be to redevelop the framework, removing the assumption that axes are perpendicular.

For non-crystalline materials, the object to be solved can be spaced from copies of itself throughout virtual space. The empty space costs extra non-zero coefficients, which gets expensive as the dimension of the space increases.

Calculations using the proposed notation for functions will run quickly if the coefficients are sparse, giving them an advantage over calculations on a grid. Analogously to grid methods, the first iterations should be run with a high zeroing threshold. The threshold can then be lowered until the desired accuracy is achieved.

A potential sticking point is modeling functions whose Taylor's series converge slowly, requiring many function-function multiplications.

Conclusion

Constructing field distributions on a Cartesian grid necessarily involves a large number of nonzero node values being calculated. Using a basis where the solutions require only a few nonzero coefficients leads to better numerical conditioning and faster computation.

Fourier series admit a succinct representation for approximating square-integrable periodic functions. These function objects can be scaled, added, differentiated, and integrated in time linear in the number of their stored coefficients; multiplication of two function objects can be accomplished in sub-quadratic time using matrix multiplication techniques.

There are other orthogonal transforms and eigenfunction bases which could be adapted to this framework. The essential requirements are that the solution functions be represented compactly, and that the ring and differential operations be easily computed on the function representations.