Dealiased Convolutions for Pseudospectral Simulations

Malcolm Roberts, University of Alberta; John C. Bowman, University of Alberta
www.math.ualberta.ca/~mroberts

Pseudospectral Simulations

The Navier–Stokes equations feature a quadratic nonlinear term which becomes a convolution in Fourier space.
In a system with m modes, the convolution has $\mathcal{O}(m^2)$ terms, which would be quite expensive to calculate directly. To avoid this cost, one makes use of the convolution theorem and fast Fourier transforms to perform the calculation in only $\mathcal{O}(m \log m)$ operations.

Dealiasing convolutions

Given the centered input vectors F_k and G_k, $k = -m+1, \dots, m-1$, we require the linear convolution

$$(F * G)_k \doteq \sum_{p=k-m+1}^{m-1} F_p G_{k-p}.$$

The FFT-based convolution treats all indices as modular, producing a cyclic convolution. The difference between the cyclic and linear convolution are called *aliasing errors*. These are removed by *zero-padding* the input arrays to length $3m$.

Higher-order convolutions

Compressible systems can result in ternary convolutions, and higher-order convolutions arise in other areas as well [1].
The n-ary convolution of the input vectors $\{F_k^i\}_{k=-m+1}^{m-1}, i = 1, \dots, n$ is

$$*\left(F^1, \dots, F^n\right)_k \doteq \sum_{k_1, \dots, k_n = -m+1}^{m-1} F_{k_1}^1 \dots F_{k_n}^n \delta_{k_1 + \dots + k_n, k}.$$

Zero-padding such a convolution requires extending the input vectors to length $(n+1)m$ in each dimension.

An n-ary Convolution $\neq n-1$ Binary Convolutions

Consider the ternary convolution $\{*(F^1, F^2, F^3)_k\}_{k=-m+1}^{m-1}$.

The term $F_{m-1}^1 F_{m-1}^2 F_{-m+1}^3$ contributes to $*(F^1, F^2, F^3)_{m-1}$.

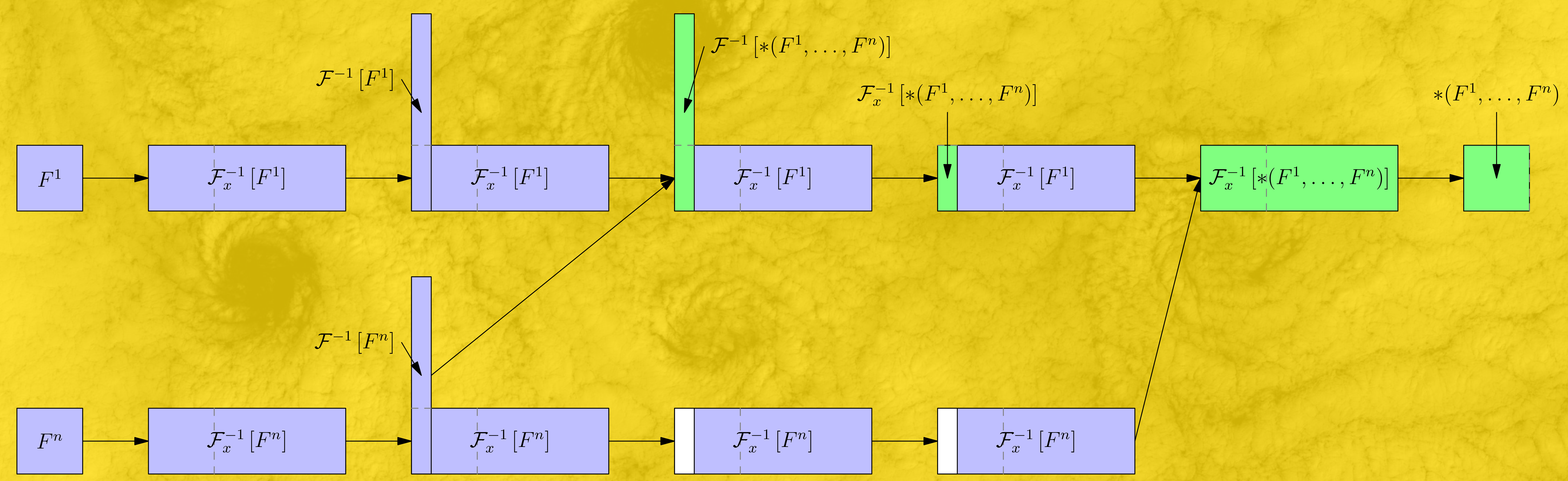
If we calculate $\{(F^1 * F^2)_k\}_{k=-m+1}^{m-1}$ then $F_{m-1}^1 F_{m-1}^2$ is discarded, so $F_{m-1}^1 F_{m-1}^2 F_{m-1}^3$ does not appear in $(F^1 * F^2) * F^3$.

Thus, for fixed-length data,

$$\boxed{*(F^1, F^2, F^3) \neq (F^1 * F^2) * F^3.}$$

Schematic Diagram of a 2D Implicitly Zero-Padded n-ary Convolution Routine

By using implicitly padded FFTs [2] and performing sub-transforms one at a time, implicitly padded convolutions can re-use work arrays. The convolution of the input arrays $\{F^1\}, \dots, \{F^n\}$ is computed as follows:



Comparison of Dealiasing Techniques

Implicitly zero-padded convolutions [2] skip or *prune* Fourier transforms which would have been performed on data which is known a priori to be zero. This is also possible with conventional explicitly padded convolutions, but is often slower in practice.

Implicit zero-padding offers a significant memory savings as compared with conventional zero padding d-dimensional convolutions:

Method	Complexity	Memory Footprint
Explicit padding	$\frac{1}{2}(n+1)^{d+1} d\, K m^d \log nm$	$\frac{n(n+1)^d}{2} m^d$
Explicit padding with pruning	$\frac{1}{2}(n+1)^2 \frac{(n+1)^d - 2^d}{n-1} K m^d \log nm$	$\frac{n(n+1)^d}{2} m^d$
Implicit padding	$\frac{1}{2}(n+1)^2 \frac{(n+1)^d - 2^d}{n-1} K m^d \log nm$	$n(n+1)2^{d-2} m^d$

Implicit zero-padding requires less memory and is less computationally complex, and can, in practice, be performed in approximately half the time of pruned explicitly padded convolutions due to decreased memory bandwidth use.

Conclusion

Implicit padding offers several advantages when calculating multidimensional convolutions:

- implicit padding requires much less memory as compared to explicit padding,
- and the computational complexity of pruned transforms,
- while being faster than explicitly padded convolutions due to decreased memory bandwidth.

Implicit padding routines are available in the open-source software library **FFTW++**, which is available at fftwpp.sourceforge.net, with libraries available for non-centered binary convolutions in one, two, and three dimensions; centered Hermitian-symmetric binary convolutions in one, two, and three dimensions; and centered Hermitian ternary convolutions in one and two dimensions, with more cases to come.

[1] John C. Bowman. Casimir cascades in two-dimensional turbulence. To appear in *J. Fluid Mech*, 2011.
[2] John C. Bowman and Malcolm Roberts. Efficient dealiased convolutions without padding. *SIAM J. Sci. Comput.*, 33(1):386406, 2011