How Important is Dealiasing for Turbulence Simulations?

John C. Bowman

University of Alberta

June 19, 2013

`www.math.ualberta.ca/~bowman/talks`

Acknowledgements: Malcolm Roberts (Aix-Marseille University)

Dealiasing

- Over 40 years ago, Orszag pointed out the importance of dealiasing in the pseudospectral method.
- Dealiasing pseudospectral convolutions, either by padding or phase-shift dealiasing is expensive.
- Some researchers in the past have therefore neglected to dealias pseudospectral simulations of strongly damped flow.
- This shortcut is typically justified with the claim that high-wavenumber damping is sufficiently strong so that the dealiasing error contributes negligibly to the large energy-containing scales.
- On the other hand, Hou and Li demonstrated in 2007 that high-order Fourier smoothing captures nearly singular solutions of the 1D inviscid Burgers equations and the 3D Euler equations more accurately and efficiently than explicit dealiasing via 2/3 zero padding.

- Given that high-Reynolds number turbulence, with well-resolved inertial ranges, falls midway between these two limiting cases of large viscosity vs. vanishing viscosity, it seems prudent to reconfirm the importance of properly dealiasing turbulence simulations. . .

A Brief History of Dealiasing

- Gauss 1866: Nachlass: Theoria interpolationis methodo nova tractata
  - Earliest example of computing Fourier transforms via divide-and-conquer strategy (FFT)
- Phillips 1959: An example of non-linear computational instability
- Cooley & Tukey 1965: An Algorithm for the Machine Calculation of Complex Fourier Series
  - popularized general FFT algorithm

- Orszag 1971: Elimination of aliasing in finite-difference schemes by filtering high-wavenumber components
    - zero padding
- Patterson & Orszag 1971: Spectral calculations of isotropic turbulence: Efficient removal of aliasing interactions
    - phase-shift dealiasing
- Choi *et al.* 1995: Parallel matrix tranpose algorithms on distributed memory concurrent computers
- Hou & Li 2007: Computing nearly singular solutions using pseudo-spectral methods
- Bowman & Roberts 2011: Efficient dealiased convolutions without padding
    - implicit dealiasing

Origin of the 2/3 Rule

1074 JOURNAL OF THE ATMOSPHERIC SCIENCES VOLUME 28

NOTES AND CORRESPONDENCE

On the Elimination of Aliasing in Finite-Difference Schemes by Filtering High-Wavenumber Components

STEVEN A. ORSZAG[1]

National Center for Atmospheric Research, Boulder, Colo.[2]

8 February 1971 and 8 April 1971

It is generally thought that, to eliminate aliasing errors in finite-difference approximations to equations with quadratic nonlinearity (e.g., the Navier-Stokes equations for incompressible flow), it is necessary to filter the top half of (each of the components of) the wavenumbers present, as first done by Phillips (1959). This is not correct. It is only necessary to filter the top one-third. If the cutoff wavenumber is K (equal to half the number of grid points in one space dimension) and if only modes k with $|k|<\frac{2}{3}K$ are allowed to be excited, then there can be no aliasing. With quadratic interaction, mode p and mode q interact to give $k=p+q$ and its aliases $k_A=p+q\pm 2K$, $p+q\pm 4K$, etc. However, for $|p|<\frac{2}{3}K$, $|q|<\frac{2}{3}K$, all the aliases satisfy $|k_A|>\frac{2}{3}K$ so that they are all filtered out. An alternative statement of the result is that it is not necessary to filter all waves with wavelengths between $2\Delta x$ and $4\Delta x$ (where Δx is the grid spacing) to eliminate aliasing. It is sufficient to filter waves with wavelengths between $2\Delta x$ and $3\Delta x$.

[1] Alfred P. Sloan Foundation Research Fellow on leave from the Department of Mathematics, M.I.T., Cambridge, Mass.

[2] The National Center for Atmospheric Research is sponsored by the National Science Foundation.

REFERENCE

Phillips, N. A., 1959: An example of nonlinear computational instability. *The Atmosphere and the Sea in Motion*, New York, Rockefeller Institute Press, p. 501.

Discrete Cyclic Convolution

- The FFT provides an efficient tool for computing the *discrete cyclic convolution*

$$\sum_{p=0}^{N-1} F_p G_{k-p},$$

where the vectors F and G have period N.

- Define the *Nth primitive root of unity:*

$$\zeta_N = \exp\left(\frac{2\pi i}{N}\right).$$

- The fast Fourier transform method exploits the properties that $\zeta_N^r = \zeta_{N/r}$ and $\zeta_N^N = 1$.

- The unnormalized *backwards discrete Fourier transform* of $\{F_k : k = 0, \ldots, N\}$ is

$$f_j \doteq \sum_{k=0}^{N-1} \zeta_N^{jk} F_k \qquad j = 0, \ldots, N-1.$$

- The corresponding *forward transform is*

$$F_k \doteq \frac{1}{N} \sum_{j=0}^{N-1} \zeta_N^{-kj} f_j \qquad k = 0, \ldots, N-1.$$

- The orthogonality of this transform pair follows from

$$\sum_{j=0}^{N-1} \zeta_N^{\ell j} = \begin{cases} N & \text{if } \ell = sN \text{ for } s \in \mathbb{Z}, \\ \dfrac{1 - \zeta_N^{\ell N}}{1 - \zeta_N^{\ell}} = 0 & \text{otherwise.} \end{cases}$$

- The pseudospectral method requires a *linear convolution*.

Convolution Theorem

$$
\begin{aligned}
\sum_{j=0}^{N-1} f_j g_j \zeta_N^{-jk} &= \sum_{j=0}^{N-1} \zeta_N^{-jk} \left(\sum_{p=0}^{N-1} \zeta_N^{jp} F_p \right) \left(\sum_{q=0}^{N-1} \zeta_N^{jq} G_q \right) \\
&= \sum_{p=0}^{N-1} \sum_{q=0}^{N-1} F_p G_q \sum_{j=0}^{N-1} \zeta_N^{(-k+p+q)j} \\
&= N \sum_{s} \sum_{p=0}^{N-1} F_p G_{k-p+sN}.
\end{aligned}
$$

- The terms indexed by $s \neq 0$ are *aliases;* we need to remove them by ensuring that $G_{k-p+sN} = 0$ whenever $s \neq 0$.

- If F_p and G_{k-p+sN} are nonzero only for $0 \leq p \leq m-1$ and $0 \leq k-p+sN \leq m-1$, then we want $k+sN \leq 2m-2$ to have no solutions for positive s.

- This can be achieved by choosing $N \geq 2m-1$.

- One can dealias by *zero padding* input data vectors of length m to length $N \geq 2m-1$:

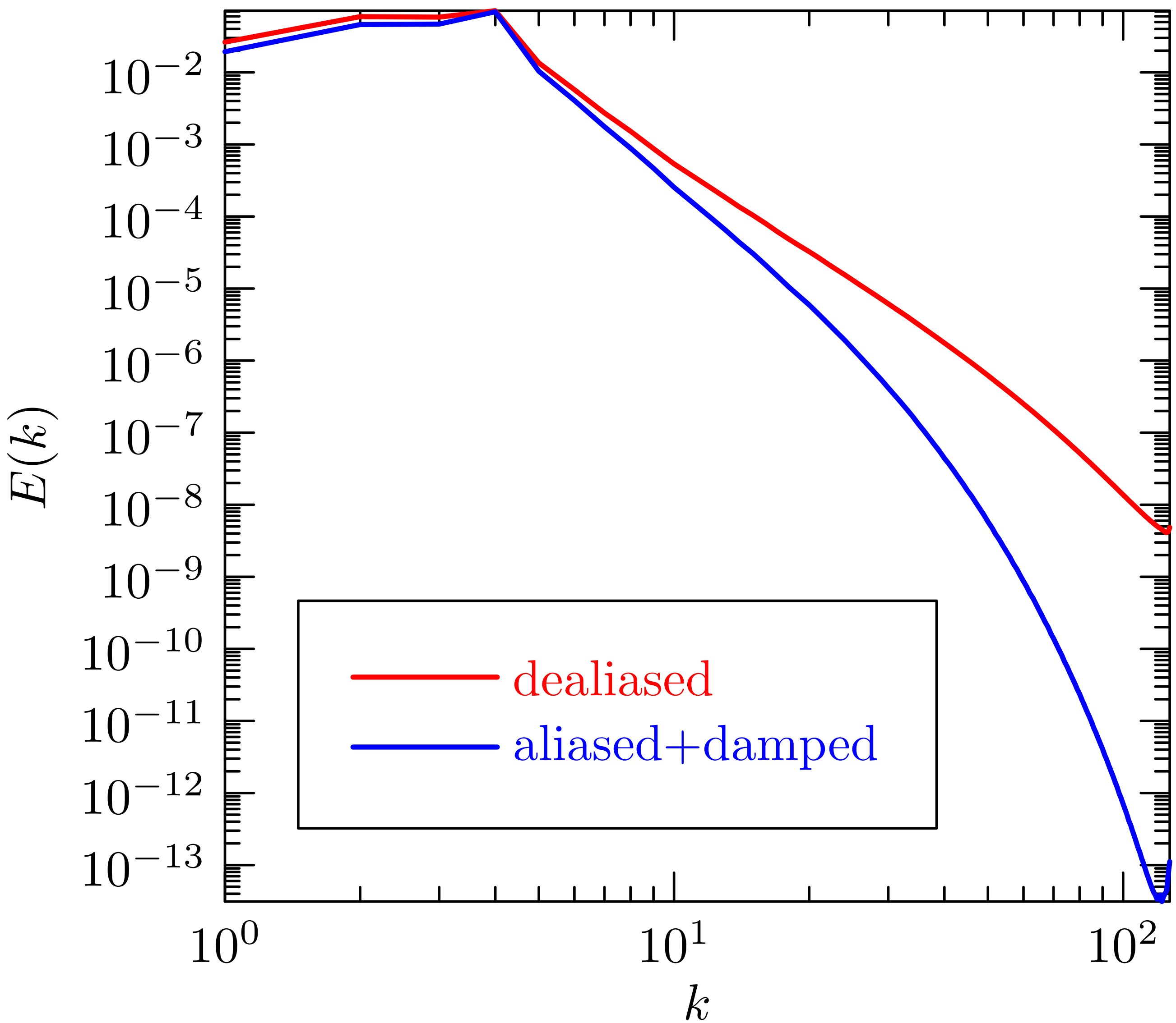
- *Explicit zero padding* prevents mode $m-1$ from beating with itself, wrapping around to contaminate mode $N = 0 \mod N$.

- Since FFT sizes with small prime factors in practice yield the most efficient implementations, the padding is normally extended to $N = 2m$.

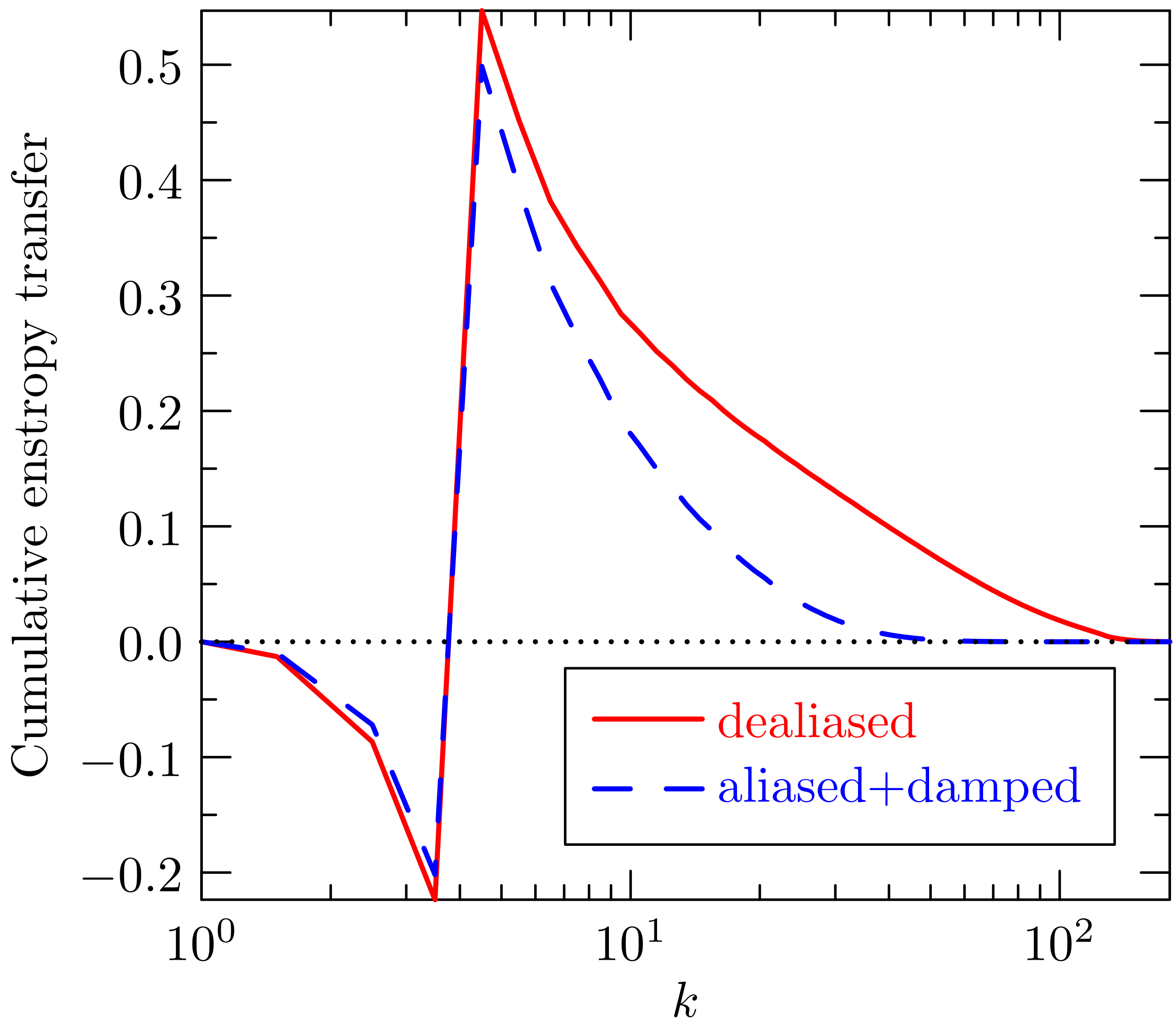
Aliasing Error

- The claim that aliasing error is negligible is problematic: in an undealiased pseudospectral simulation on $[-N/2, N/2]$, the inertial range mode at $N/3$ will beat with itself to generate a spurious harmonic at wavenumber $2N/3 = -N/3 (\mathrm{mod}\, N)$.
- Moreover, as Phillips pointed out in 1959, aliasing errors not only contaminate the largest scales, they quickly lead to high-wavenumber explosive numerical instability.
- In one experiment, we had to increase the viscosity by a factor of 15 to damp out the aliasing instability.
- This additional damping completely destroyed the inertial range and modified the energy transfers and large-scale energy spectrum.

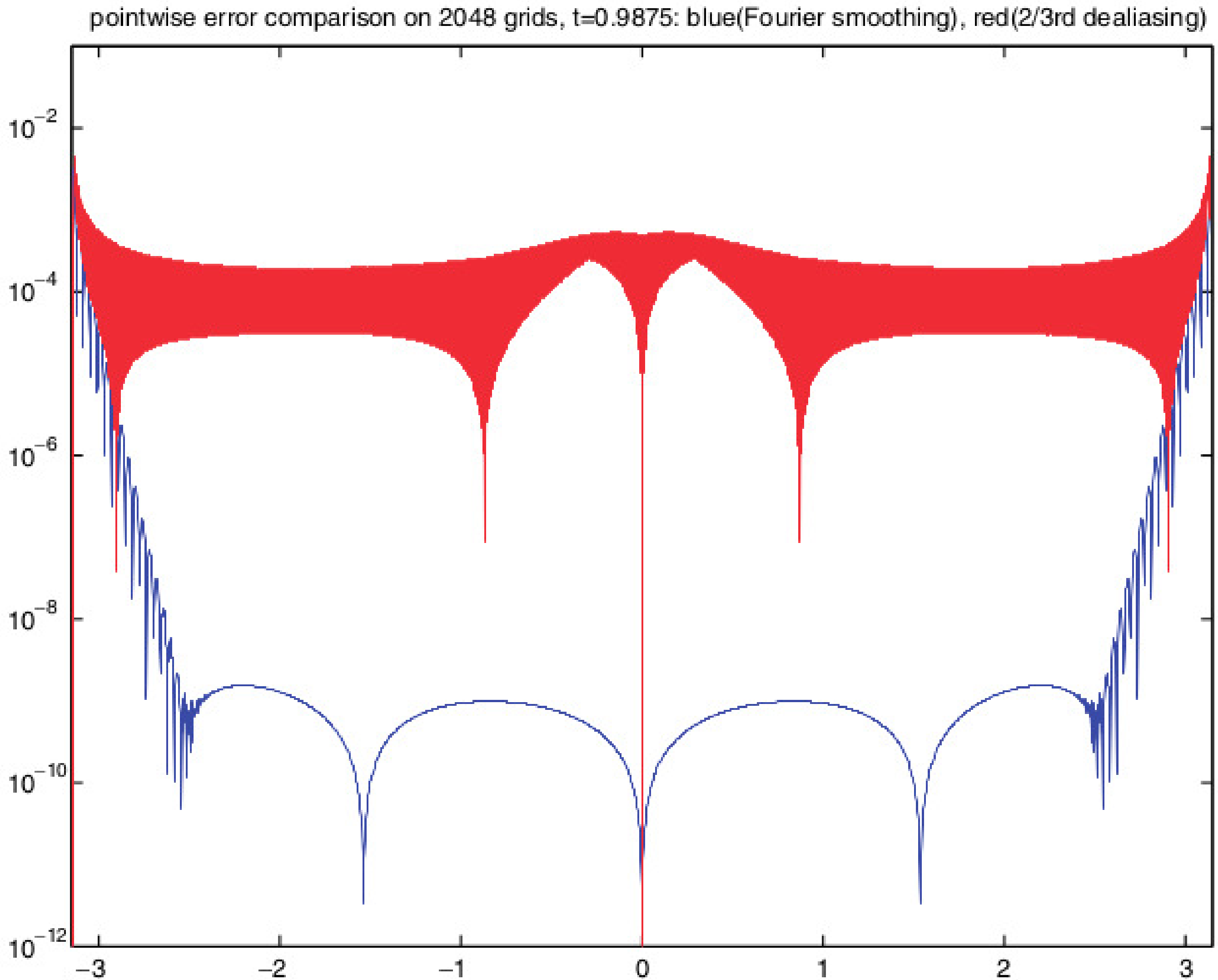
2D Enstrophy Cascade



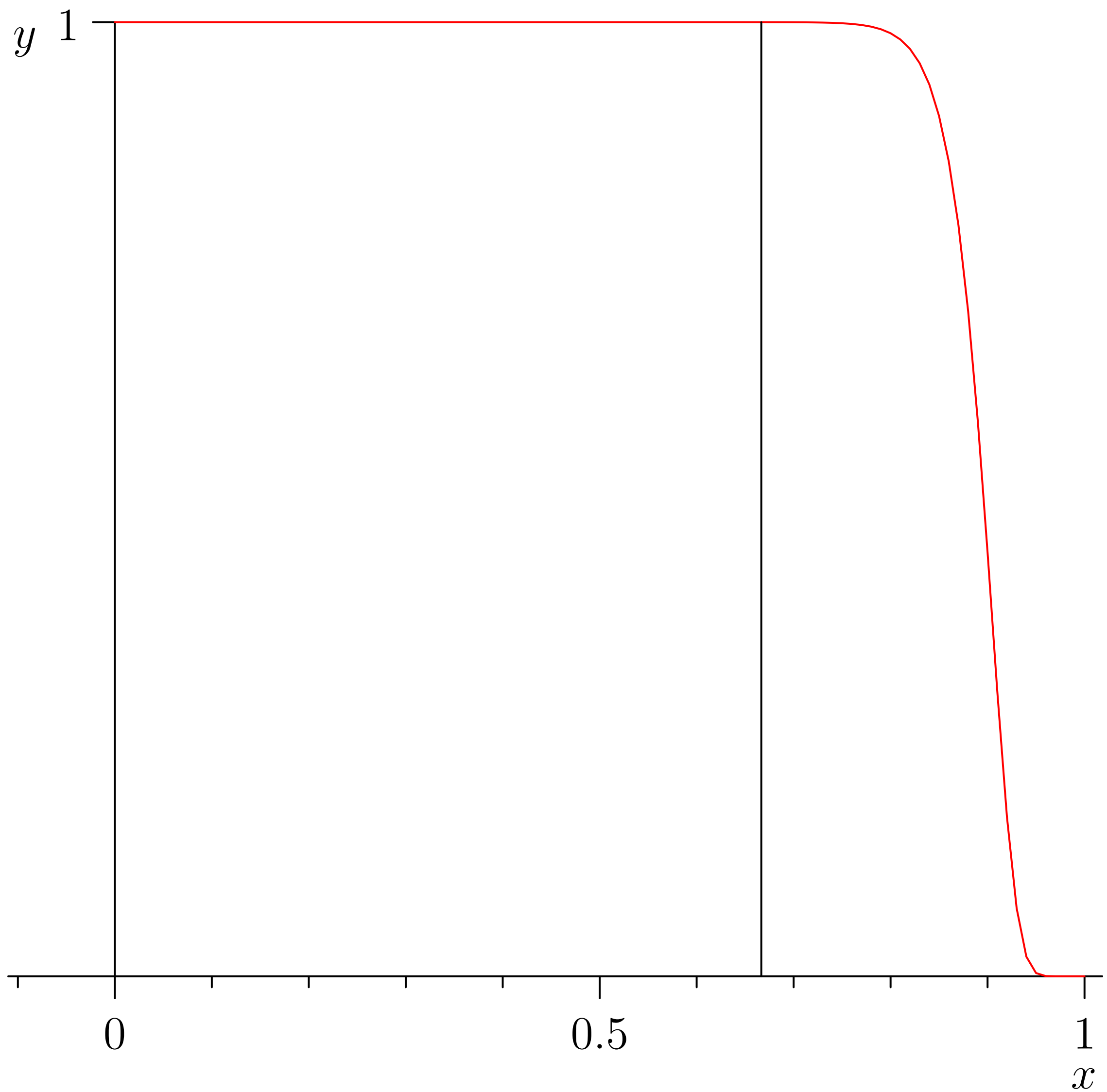
2D Enstrophy Transfer



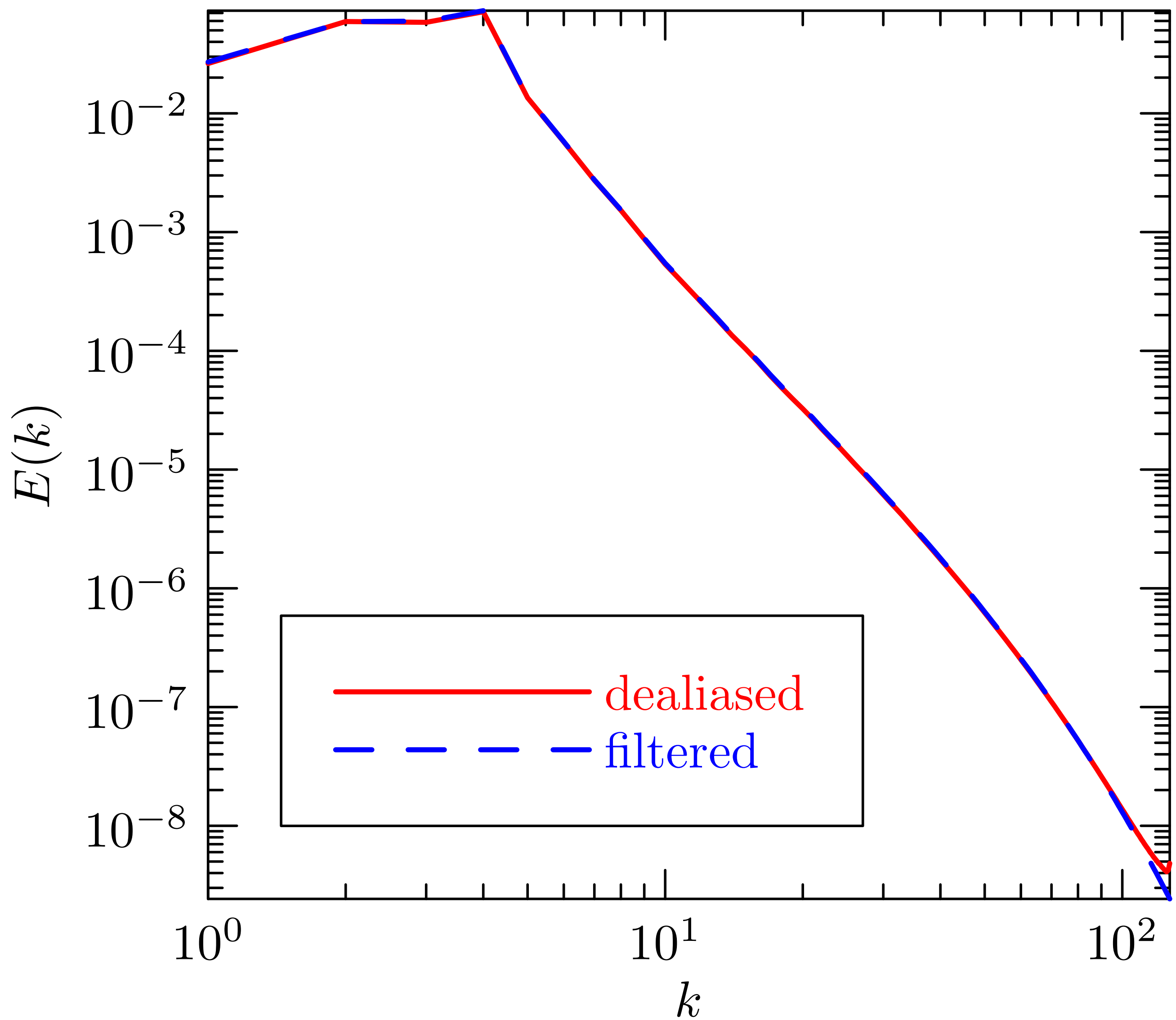
Partial Dealiasing with a Fourier Filter



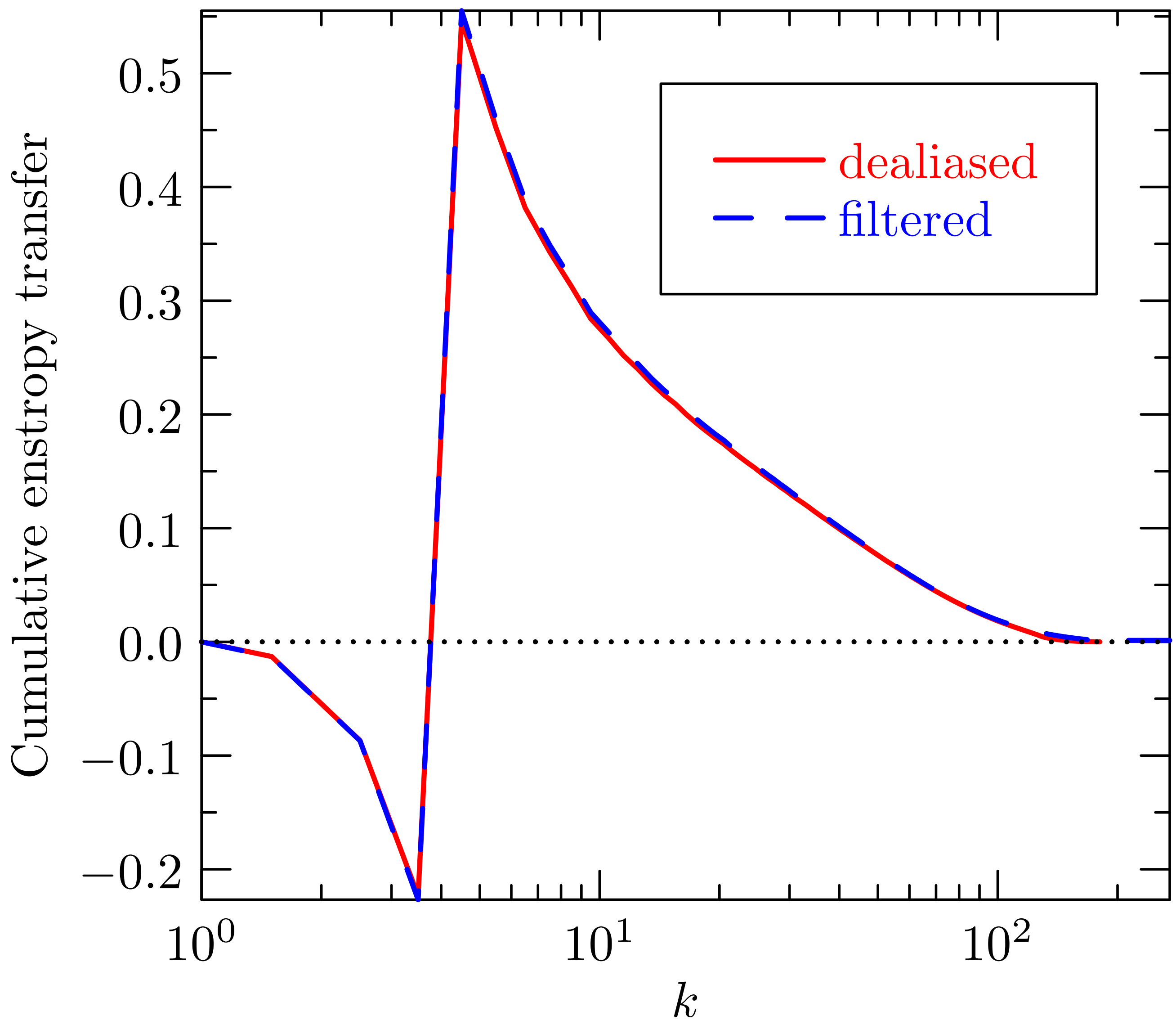
Fourier Filter $y = e^{-36x^{36}}$



2D Enstrophy Cascade



2D Enstrophy Transfer



- The recent introduction of implicit dealiasing, which in two and three dimensions are roughly twice as fast as explicit dealiasing, offsets the claim that smoothing via a Fourier filter is 20% more efficient than dealiasing…

Implicit Padding

- Let $N = 2m$. For $j = 0, \ldots, 2m-1$ we want to compute

$$f_j = \sum_{k=0}^{2m-1} \zeta_{2m}^{jk} F_k.$$

- If $F_k = 0$ for $k \geq m$, one can easily avoid looping over the unwanted zero Fourier modes by decimating in wavenumber:

$$f_{2\ell} = \sum_{k=0}^{m-1} \zeta_{2m}^{2\ell k} F_k = \sum_{k=0}^{m-1} \zeta_m^{\ell k} F_k,$$

$$f_{2\ell+1} = \sum_{k=0}^{m-1} \zeta_{2m}^{(2\ell+1)k} F_k = \sum_{k=0}^{m-1} \zeta_m^{\ell k} \zeta_{2m}^{k} F_k, \qquad \ell = 0, 1, \ldots m-1.$$

- This requires computing two subtransforms, each of size m, for an overall computational scaling of order $2m \log_2 m = N \log_2 m$.

- Odd and even terms of the convolution can then be computed separately, multiplied term-by-term, and transformed again to Fourier space:

$$
\begin{aligned}
2mF_k &= \sum_{j=0}^{2m-1} \zeta_{2m}^{-kj} f_j \\
&= \sum_{\ell=0}^{m-1} \zeta_{2m}^{-k2\ell} f_{2\ell} + \sum_{\ell=0}^{m-1} \zeta_{2m}^{-k(2\ell+1)} f_{2\ell+1} \\
&= \sum_{\ell=0}^{m-1} \zeta_{m}^{-k\ell} f_{2\ell} + \zeta_{2m}^{-k} \sum_{\ell=0}^{m-1} \zeta_{m}^{-k\ell} f_{2\ell+1} \qquad k = 0, \ldots, m-1.
\end{aligned}
$$

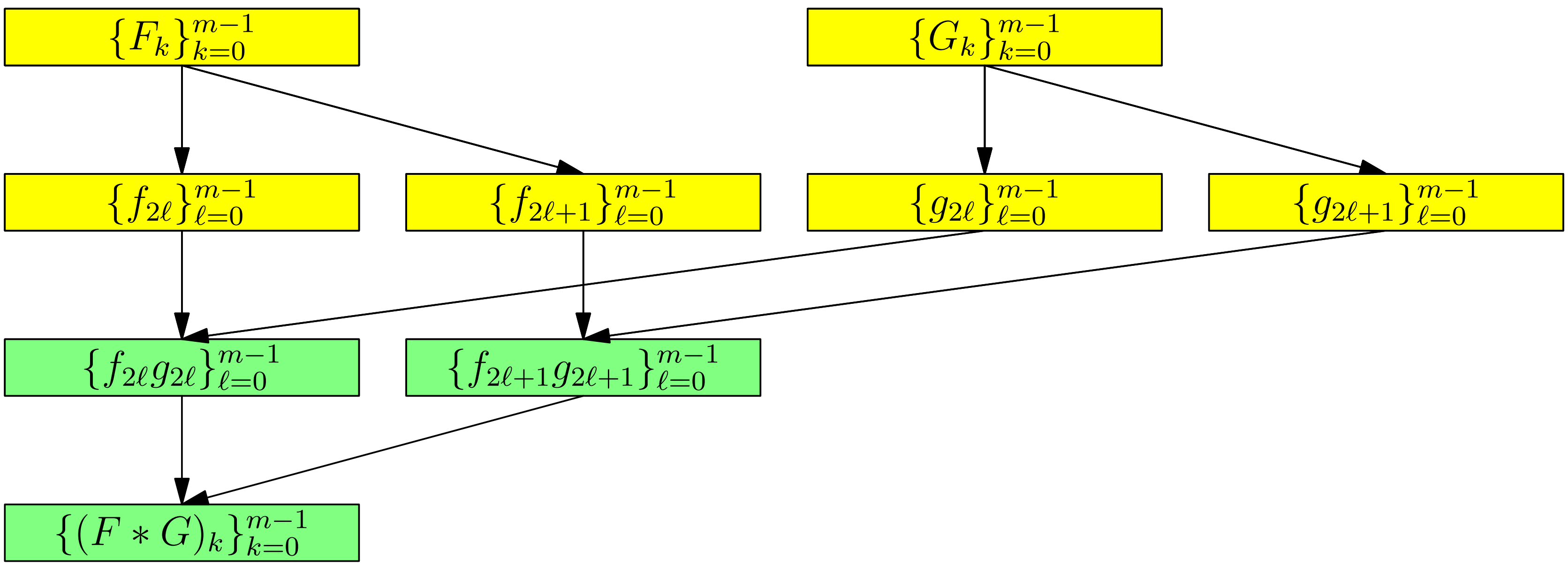
- No bit reversal is required at the highest level.

- An implicitly padded convolution is implemented as in our `FFTW++` library (version 1.13) as `cconv(f,g,u,v)` computes an in-place implicitly dealiased convolution of two complex vectors `f` and `g` using two temporary vectors `u` and `v`, each of length m.

- This in-place convolution requires six out-of-place transforms, thereby avoiding bit reversal at all levels.

- The computational complexity is $6Km\log_2 m$.

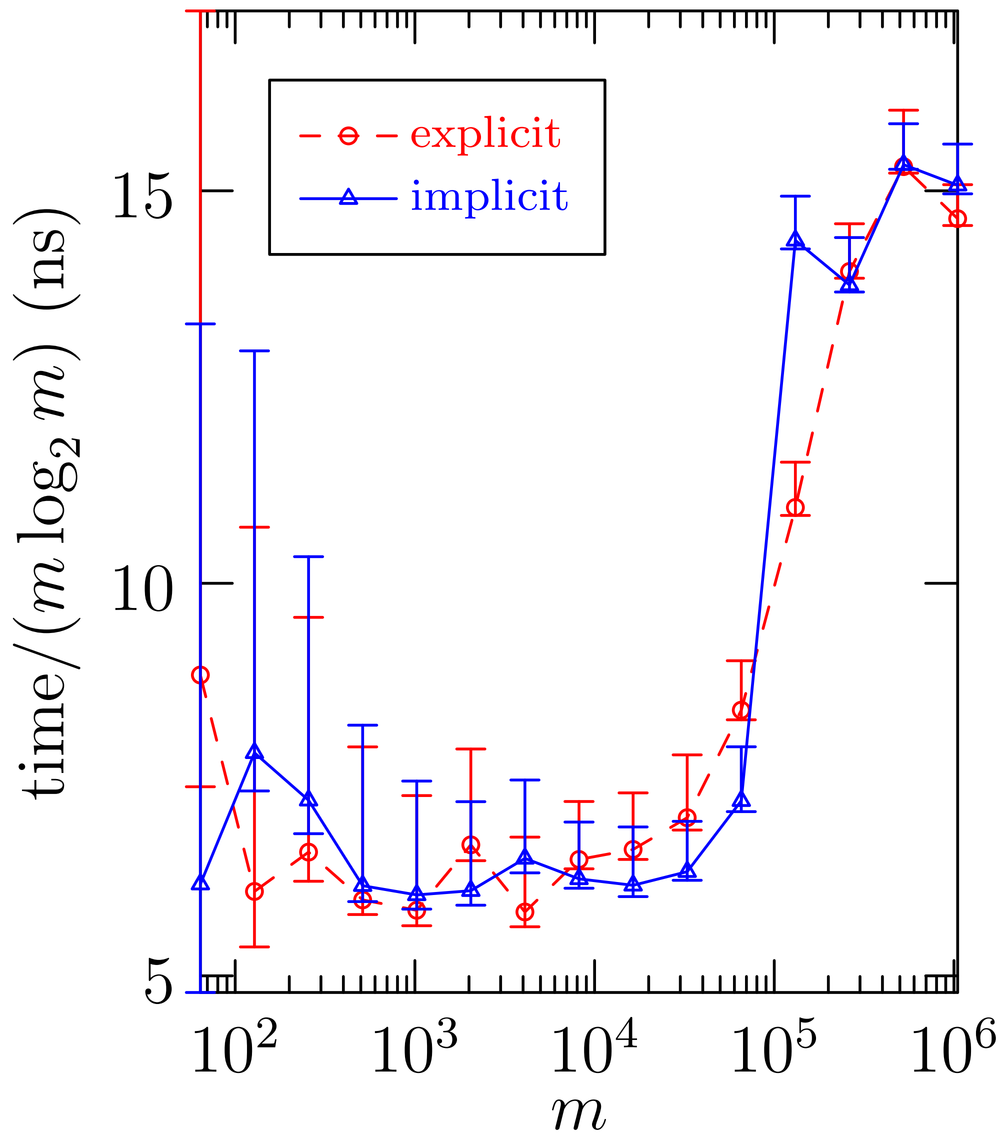
- The numerical error is similar to explicit padding.



Input: vector f, vector g
Output: vector f

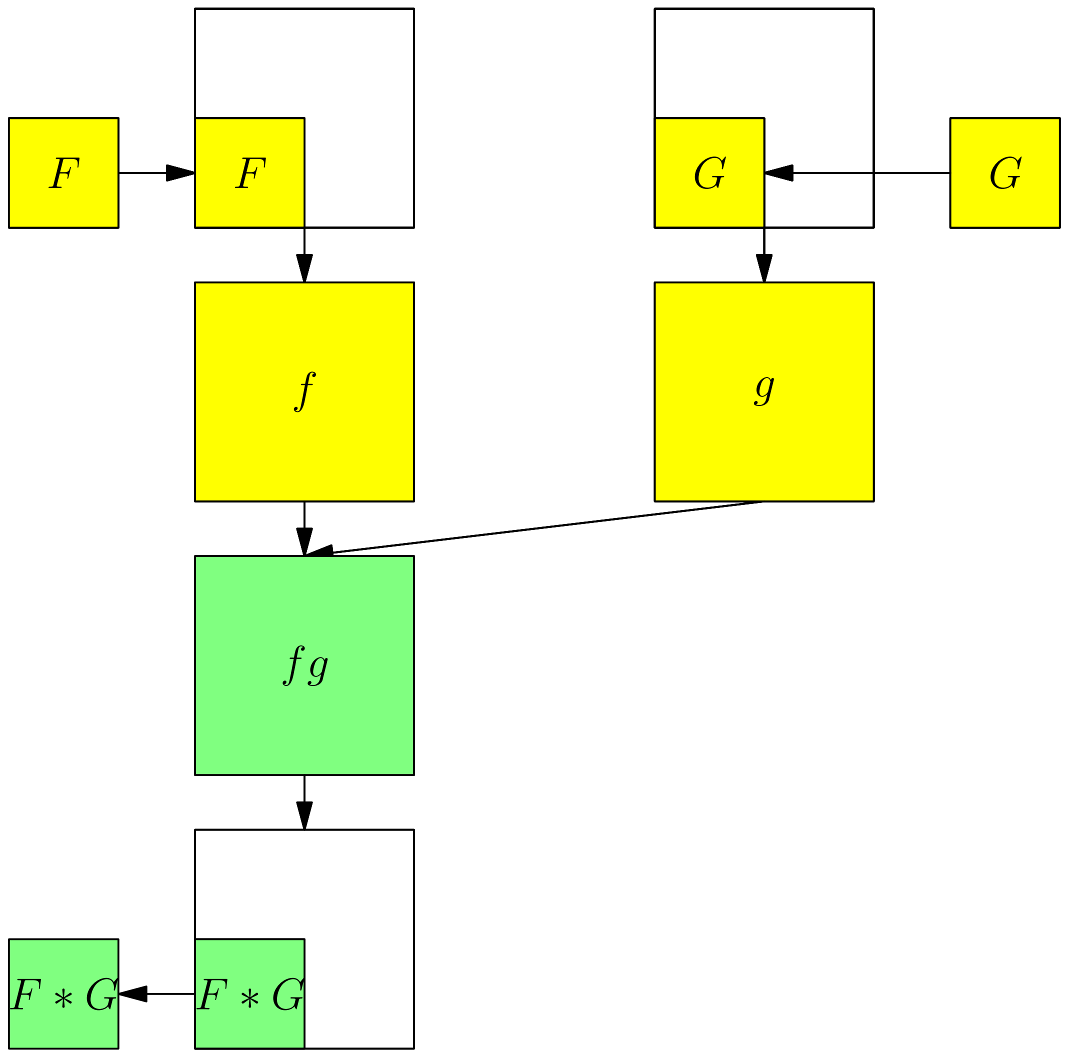
```
u ← fft^{-1}(f);
v ← fft^{-1}(g);
u ← u * v;
for k = 0 to m − 1 do
    f[k] ← ζ_{2m}^k f[k];
    g[k] ← ζ_{2m}^k g[k];
end
v ← fft^{-1}(f);
f ← fft^{-1}(g);
v ← v * f;
f ← fft(u);
u ← fft(v);
for k = 0 to m − 1 do
    f[k] ← f[k] + ζ_{2m}^{-k} u[k];
end
return f/(2m);
```

Implicit Padding in 1D



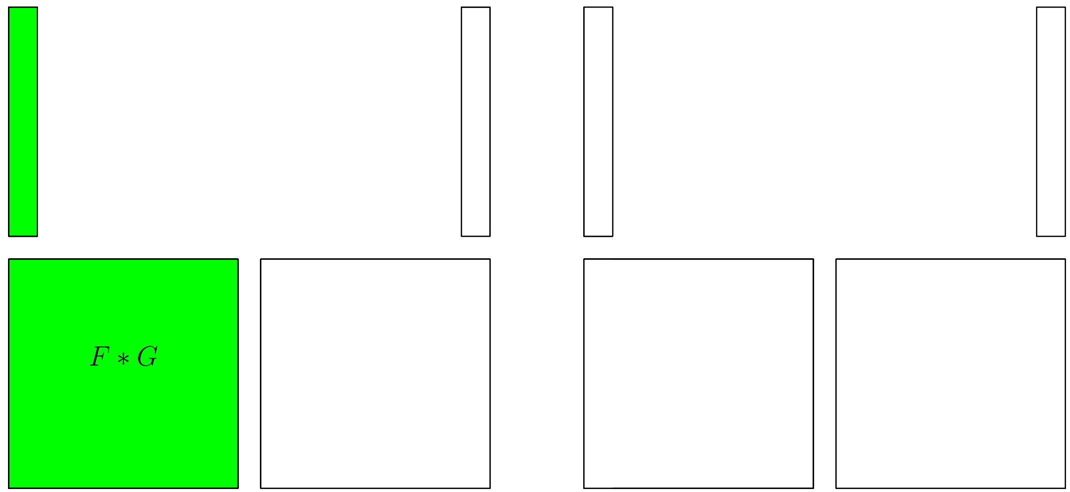
Convolutions in Higher Dimensions

- An explicitly padded convolution in 2 dimensions requires 12 padded FFTs, and 4 times the memory of a cyclic convolution.

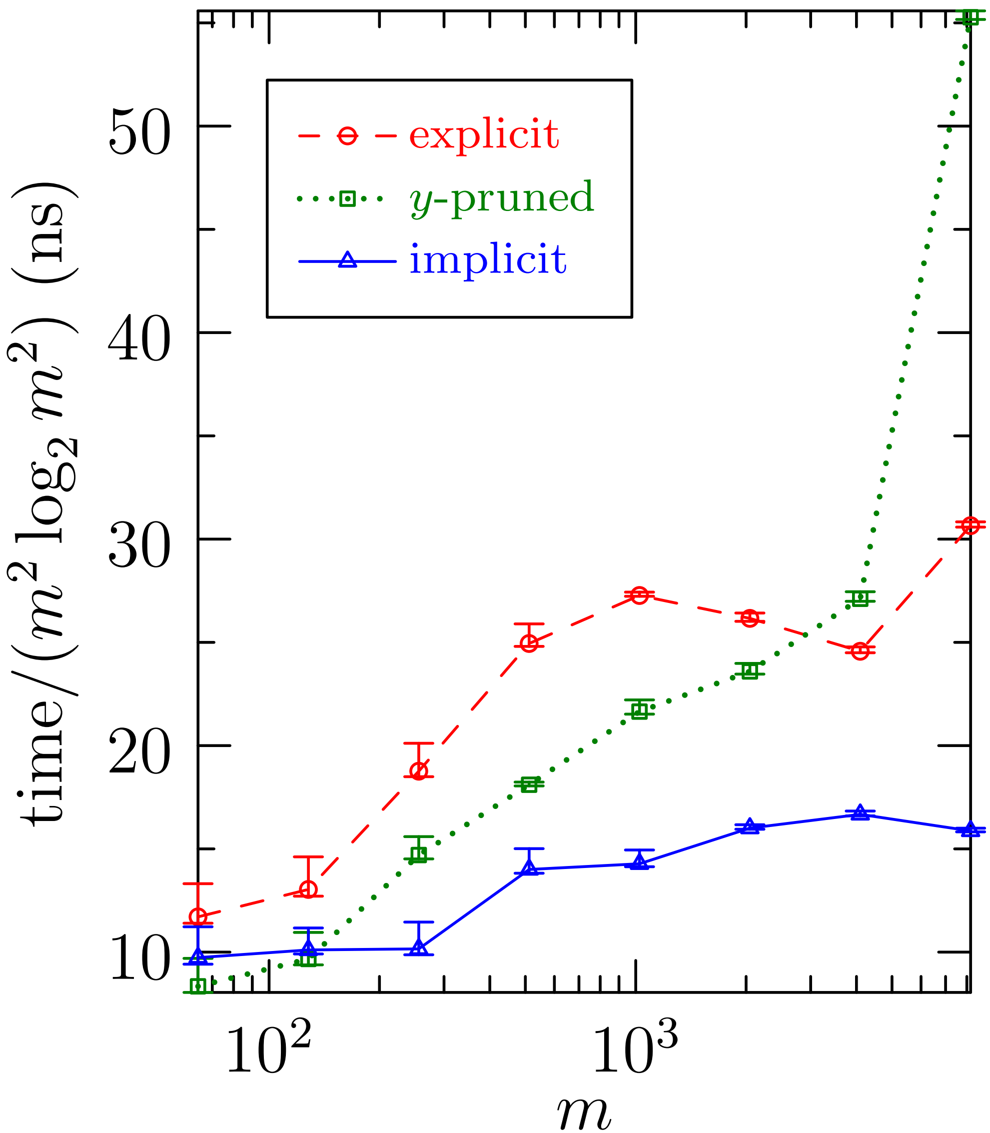


Implicit Padding in 2D

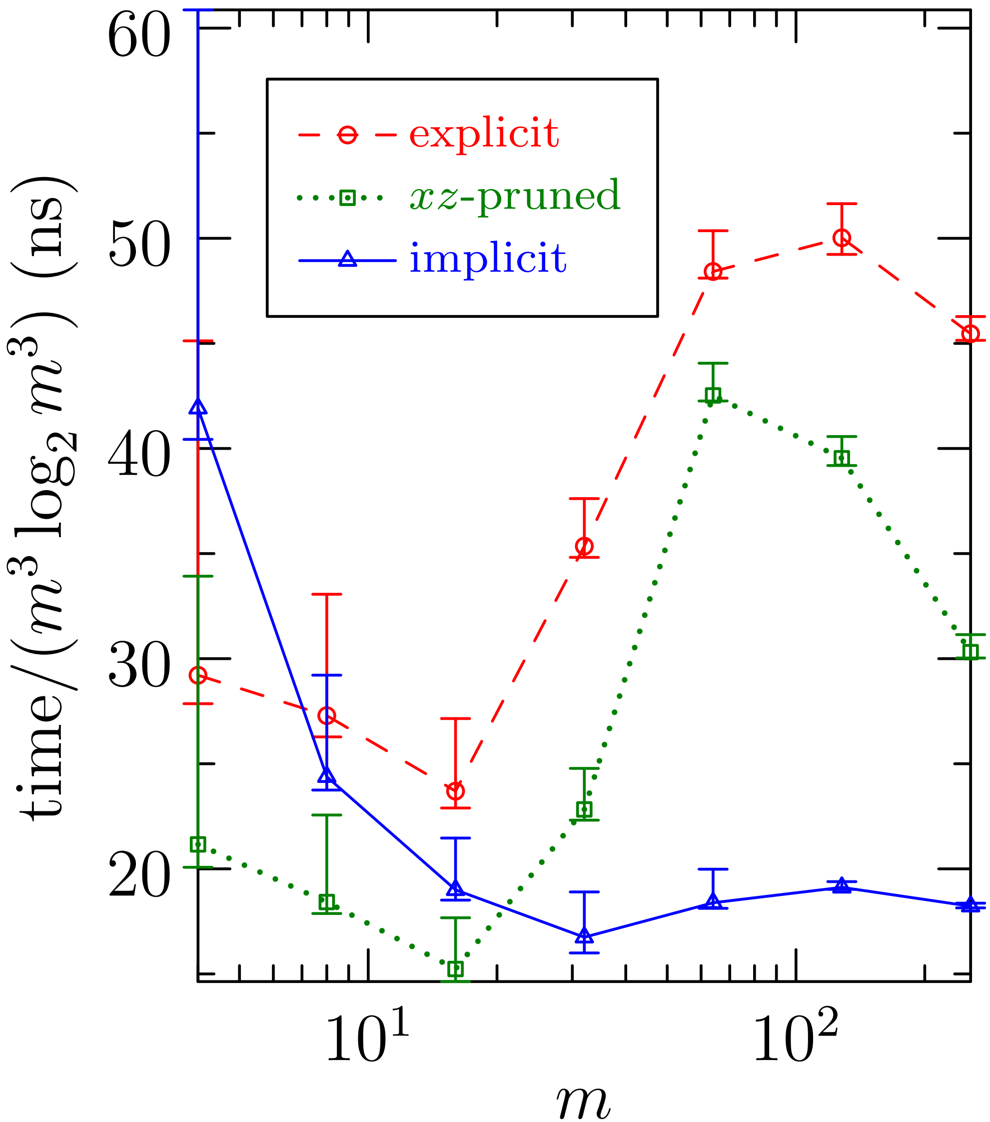
- Extra work memory need not be contiguous with the data.



Implicit Padding in 2D



Implicit Padding in 3D



Hermitian Convolutions

- *Hermitian convolutions* arise when the input vectors are Fourier transforms of real data:

$$f_{N-k} = \overline{f_k}.$$

Centered Convolutions

- For a *centered convolution,* the Fourier origin $(k = 0)$ is centered in the domain:

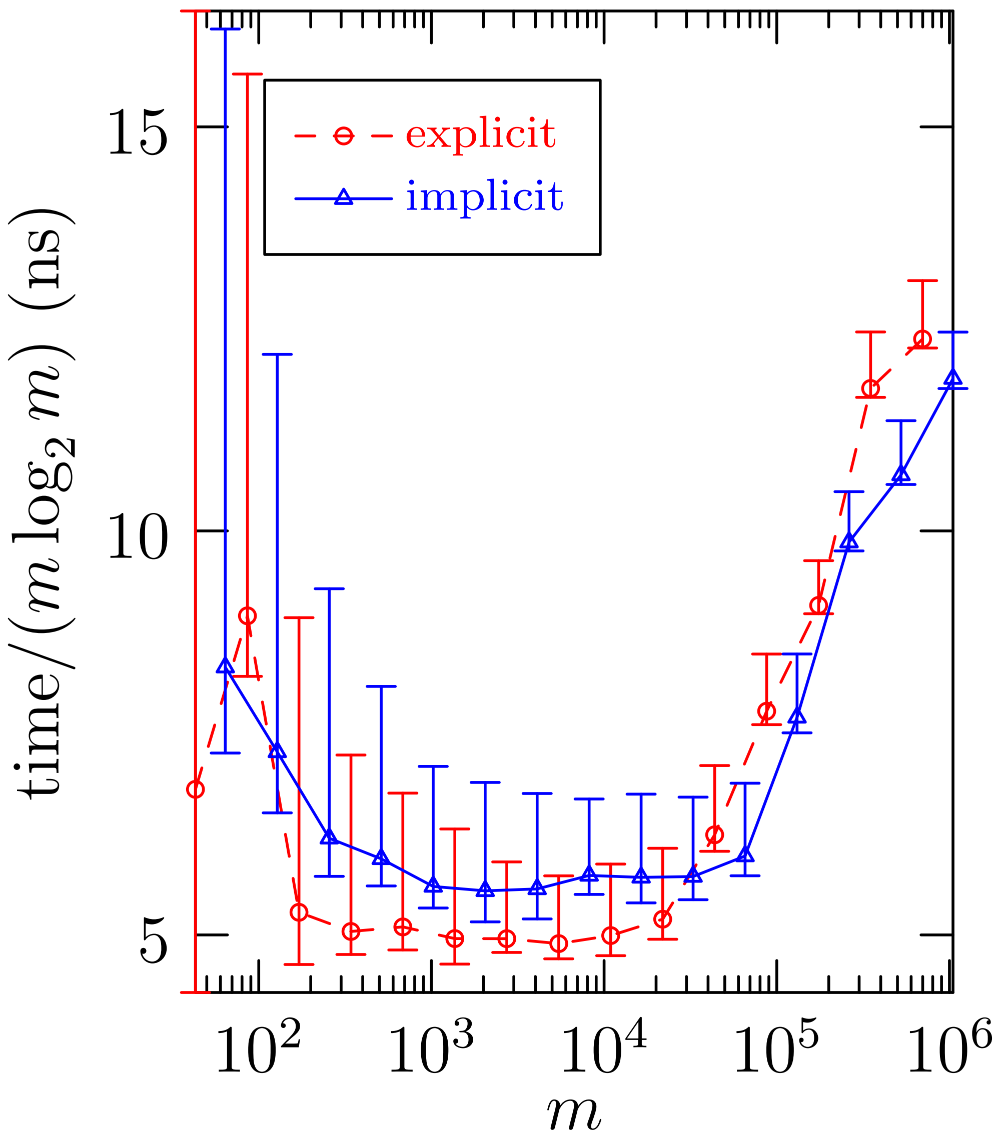
$$\sum_{p=k-m+1}^{m-1} f_p g_{k-p}$$

- Here, one needs to pad to $N \geq 3m-2$ to prevent mode $m-1$ from beating with itself to contaminate the most negative (first) mode, corresponding to wavenumber $-m+1$. Since the ratio of the number of physical to total modes, $(2m-1)/(3m-2)$ is asymptotic to $2/3$ for large m, this padding scheme is often referred to as the *2/3 padding rule.*

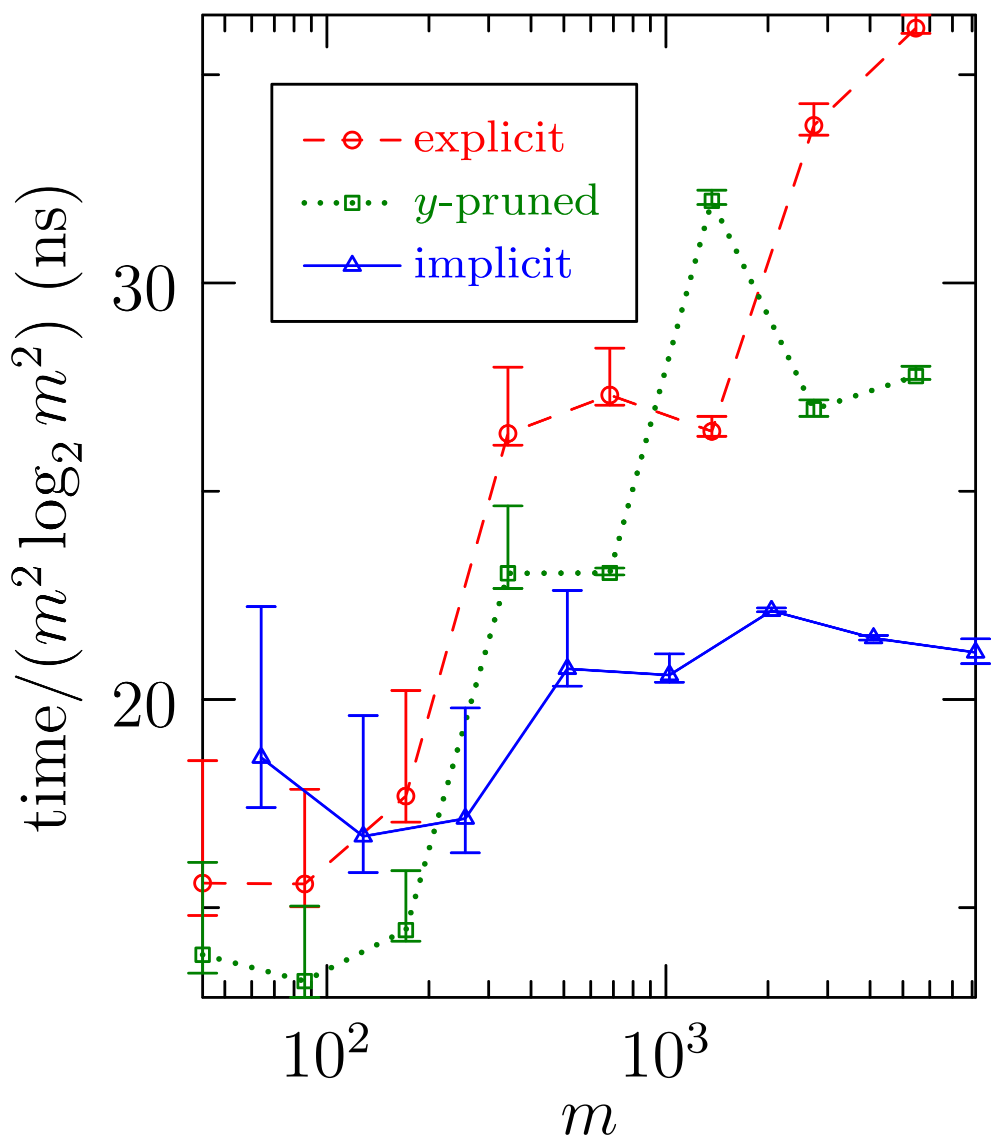
- The Hermiticity condition then appears as

$$f_{-k} = \overline{f_k}.$$

Implicit Hermitian Centered Padding in 1D



Implicit Hermitian Centered Padding in 2D



Parallelization

- Our implicit and explicit convolution routines have been multithreaded for shared-memory architectures.
- Parallel generalized slab/pencil model implementations have recently been developed for distributed-memory architectures (available in svn repository and upcoming 1.14 release).
- The key bottleneck is the distributed matrix transpose.
- We have compared several distributed matrix transpose algorithms, both blocking and nonblocking, under both pure MPI and hybrid MPI/OpenMP architectures.
- Local transposition is not required within a single MPI node.
- Another advantage of hybrid MPI/OpenMP over pure MPI for matrix transposition is that it yields a larger communication block size.

- Hybrid MPI/OpenMP is more efficient (by roughly a factor of 2) than pure MPI for distributed matrix transposes.

- However, FFTs seem to be more efficiently calculated with pure MPI.

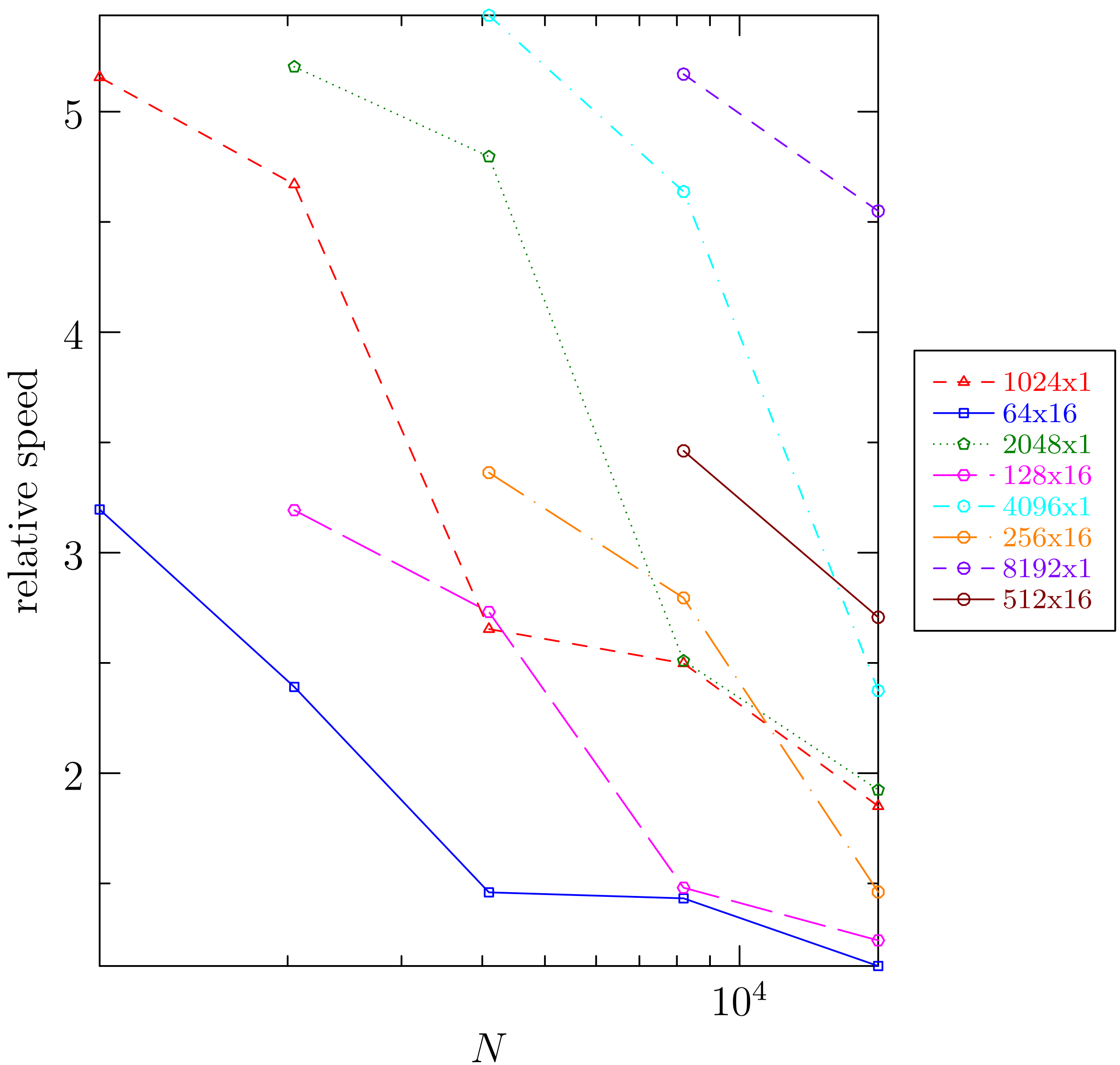
- Nevertheless, hybrid MPI/OpenMP offers some advantages:

  – smaller problems sizes to be distributed over a large number of processors;

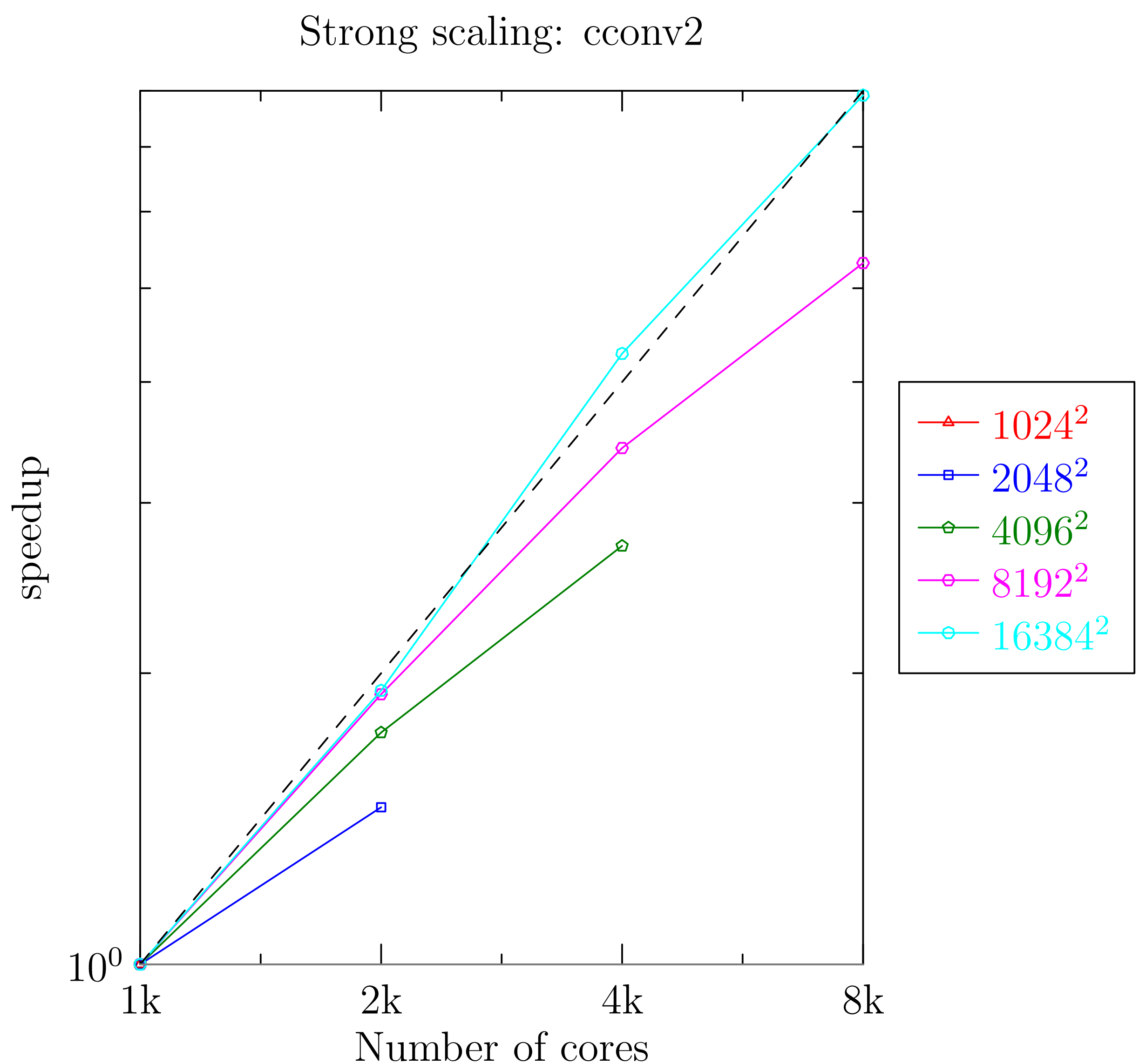
  – more slab-like than pencil-like model; this reduces the size of or even eliminates the need for the second transpose.

- Our attempts thus far to overlap communication and computation have unfortunately resulted in fragmented communication, hurting performance.

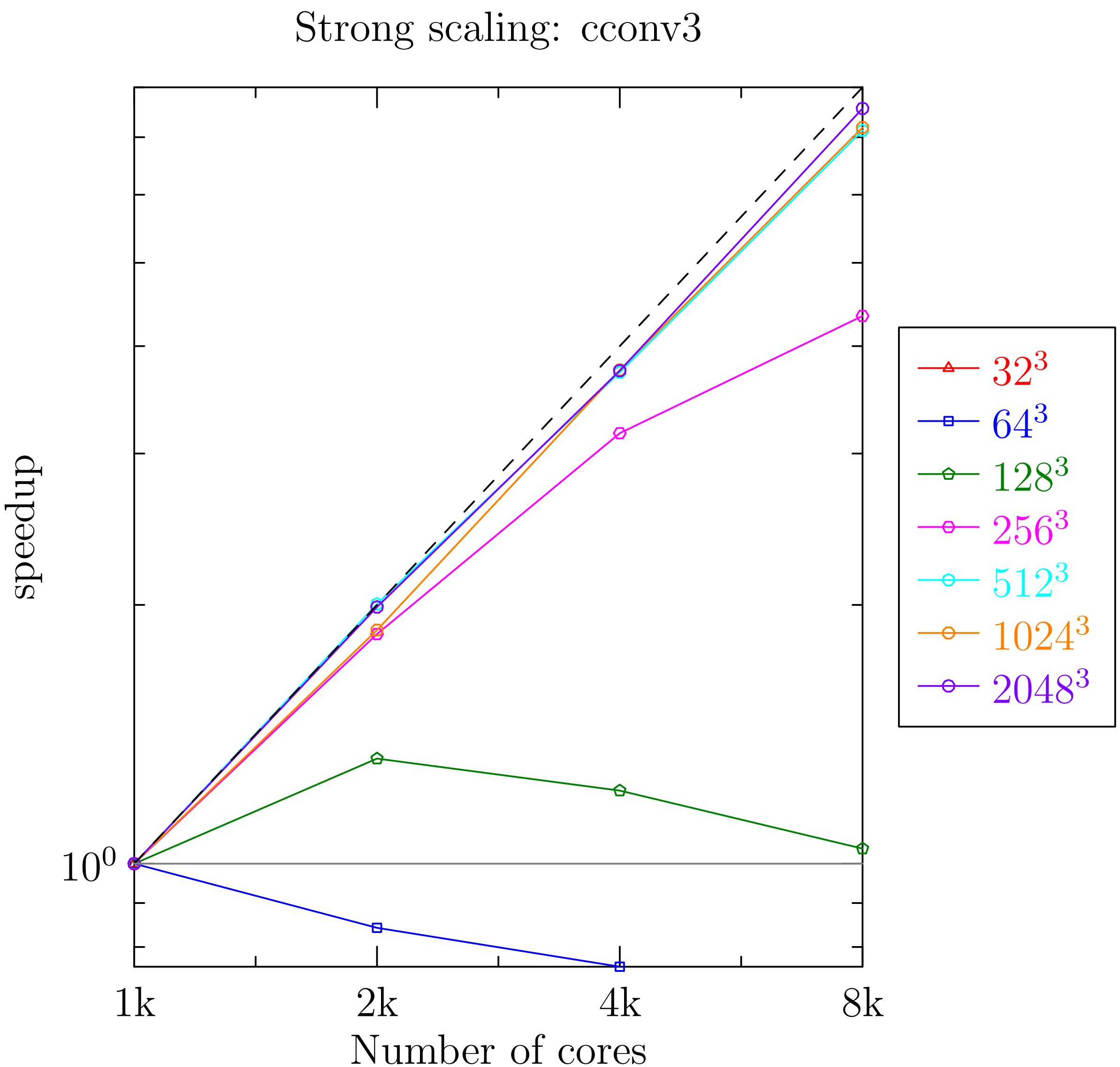
Pure MPI Speedup of 2D Implicit vs. Explicit Convolutions for Different Problem Sizes



Pure MPI Scaling of 2D Implicit Convolutions



Pure MPI Scaling of 3D Implicit Convolution



Conclusions

- Dealiasing of pseudospectral simulations is essential for stability and accuracy.
- For turbulence simulations, partial dealiasing via a Fourier filter seems neither necessary nor desirable.
- With the advent of implicit dealiasing, partial Fourier filtering is no longer the most efficient option.
- Memory savings: in d dimensions implicit padding asymptotically uses $1/2^{d-1}$ of the memory require by conventional explicit padding.
- The factor of 2 in computational savings is due to increased data locality.
- Highly optimized and parallelized implicit dealiasing routines have been implemented as a software layer **`FFTW++`** on top of the **`FFTW`** library and released under the Lesser GNU Public License: **`http://fftwpp.sourceforge.net/`**

References

[Bowman & Roberts 2011] J. C. Bowman & M. Roberts, SIAM J. Sci. Comput., **33**:386, 2011.

[Cooley & Tukey 1965] J. W. Cooley & J. W. Tukey, Mathematics of Computation, **19**:297, 1965.

[Gauss 1866] C. F. Gauss, "Nachlass: Theoria interpolationis methodo nova tractata," in *Carl Friedrich Gauss Werke*, volume 3, pp. 265–327, Königliche Gesellschaft der Wissenschaften, Göttingen, 1866.

[Hou & Li 2007] T. Y. Hou & R. Li, Journal of Computational Physics, **226**:379, 2007.

[Orszag 1971] S. A. Orszag, Journal of the Atmospheric Sciences, **28**:1074, 1971.

[Patterson Jr. & Orszag 1971] G. S. Patterson Jr. & S. A. Orszag, Physics of Fluids, **14**:2538, 1971.

[Phillips 1959] N. A. Phillips, The Atmosphere and the Sea in Motion, p. 501, 1959.