Documentation

Accelerate / Fast Fourier transforms / In-Place Functions for 1D Complex FFT

API Collection

In-Place Functions for 1D Complex FFT

Perform fast Fourier transforms in place on 1D complex data.

Overview

The functions in this group use the following operation for a complex-to-complex transform:

```
N = 1 << Log2N;

scale = 0 < Direction ? 1 : 1./N;

// Define a complex vector, h:
for (j = 0; j < N; ++j)
    h[j] = C->realp[j*IC] + i * C->imagp[j*IC];

// Perform Discrete Fourier Transform.
for (k = 0; k < N; ++k)
    H[k] = scale * sum(h[j] * e**(-Direction*2*pi*i*j*k/N), 0 <= j < N);

// Store result.
for (k = 0; k < N; ++k)
{
    C->realp[k*IC] = Re(H[k]);
    C->imagp[k*IC] = Im(H[k]);
}
```

The temporary buffer versions perform the same operation but use a temporary buffer for improved performance.

Topics

In-Place FFT Functions

`vDSP_fft_zip`

Computes a forward or inverse in-place, single-precision complex FFT.

`vDSP_fft_zipD`

Computes a forward or inverse in-place, double-precision complex FFT.

In-Place FFT Functions with Temporary Buffer

`vDSP_fft_zipt`

Computes a forward or inverse in-place, single-precision complex FFT using a temporary buffer.

`vDSP_fft_ziptD`

Computes a forward or inverse in-place, double-precision complex FFT using a temporary buffer.

See Also

Functions for 1D Complex FFT

☰ Out-of-Place Functions for 1D Complex FFT

Perform fast Fourier transforms out of place on 1D complex data.