Optimizing FFT Performance

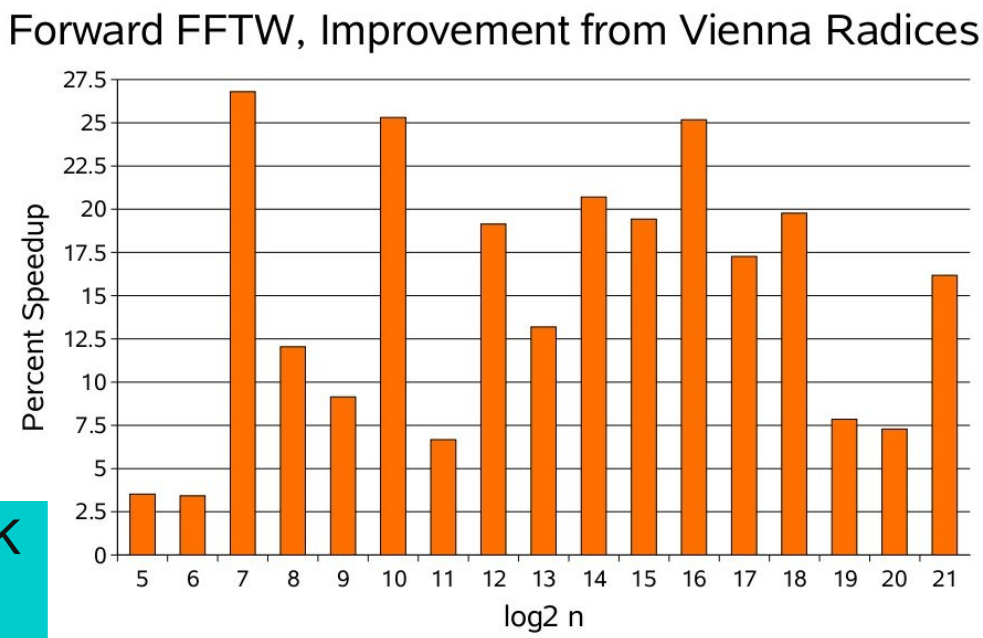
- Samara Technology Group, LLC (STG) was tasked to improve FFT performance on the SiCortex MIPS based 5,832 core cluster computer using the open source FFTW package.
- FFTW is a subroutine library for computing the discrete Fourier transform (DFT) in one or more dimensions, of arbitrary input size, and of both real and complex data. It is widely accepted and a defacto standard.
- Companies choose FFTW because it quickly provides acceptable compiled FFT performance; none the less it did not meet expectations on the SiCortex platform.
- The most important optimization case boosts total system performance by a half a teraflop.

The Discrete Fourier Transform is referred to as the “Fast Fourier Transform” following a set of algorithms popularized by Cooley and Tukey which can reduce the calculation time by several orders of magnitude. Perhaps the single most important algorithmic domain in all of Computer Science, the FFT finds uses in devices ranging from the smallest appliance to the worlds largest HPC systems.

Benefit from the Numerics Research Community

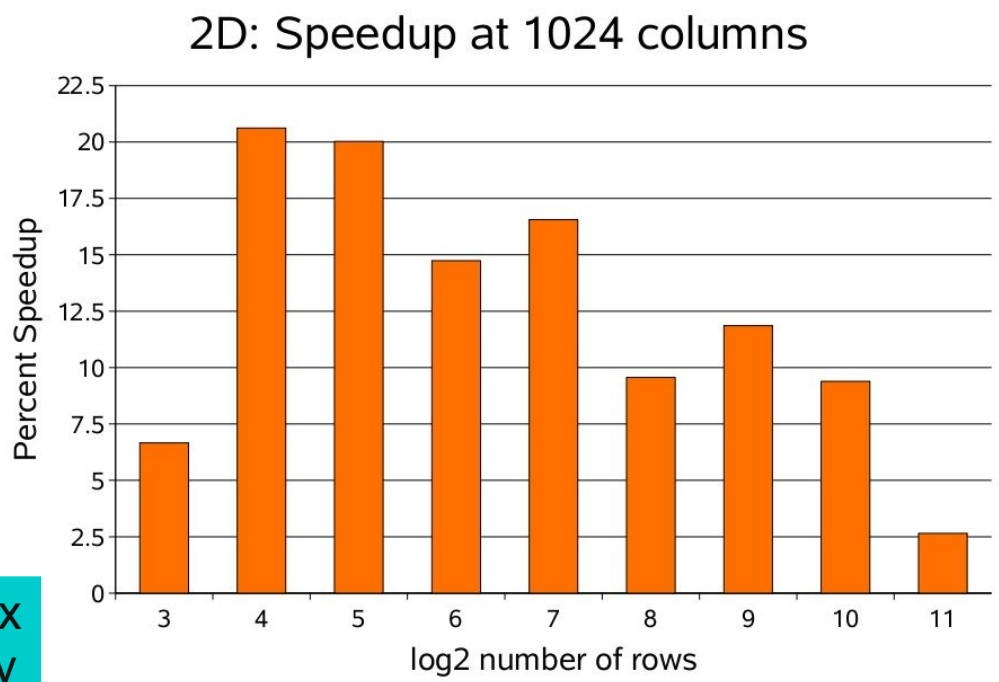
- Investigations focused on FFT research from the Technical University of Vienna.
- Alternate radix algebraic expansion provided more balanced multiply/add ratio, leading to better total throughput on combined multiply-add architectures.
- MIPS internal rounding permits this approach without loss of precision when compared to traditional radix expansion.
- Vienna work was extended to apply to larger range of radices.
- FFTW “codelets” were then replaced with the modified radix code and the package rebuilt.
- Collectively this work is referred to as the “Vienna Radices” or “Vienna Modifications.”

Percent Benefit per Length, 1D Complex FFT



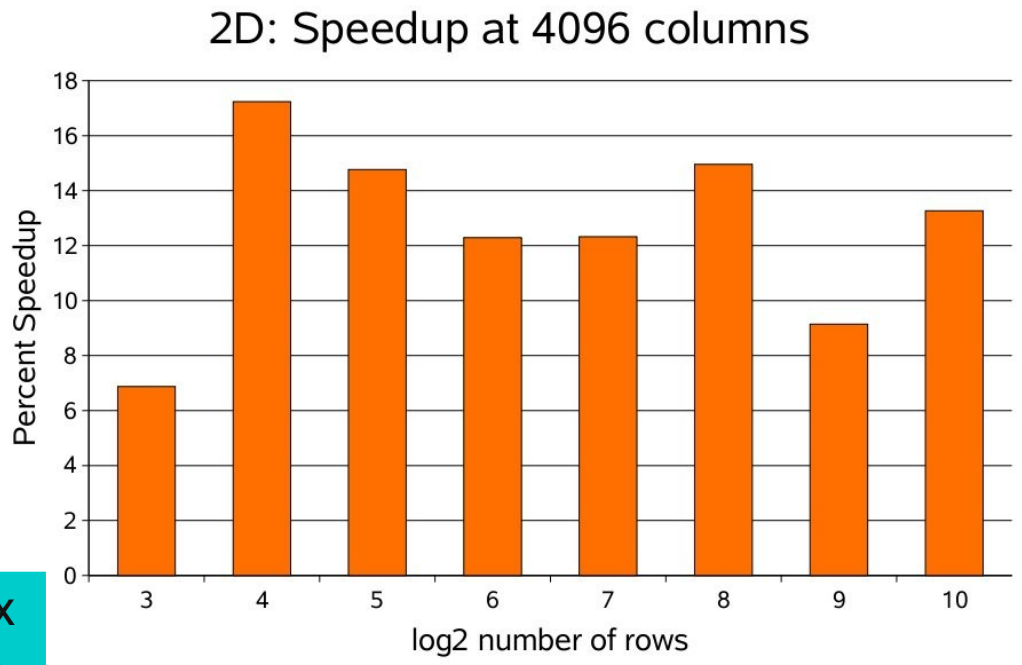
Key benchmark 1K forward complex FFT shows 25% improvement over unmodified FFTW

Percent Benefit per Length, 2D Complex FFT



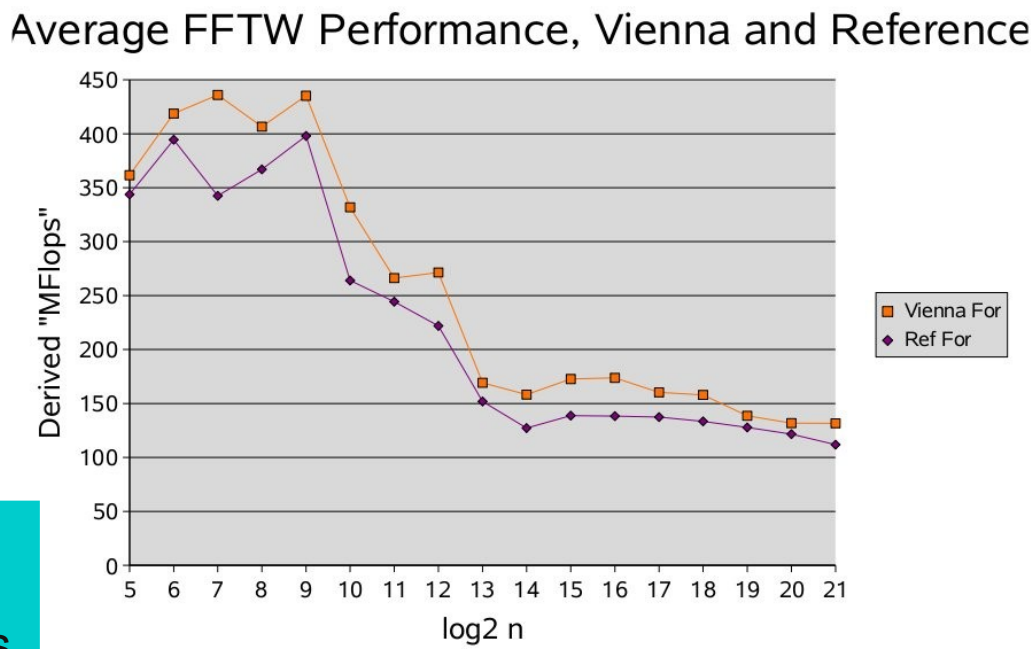
1K by 1K complex FFT shows nearly 10% speedup over unmodified FFTW

Percent Benefit per Length, 2D Complex FFT



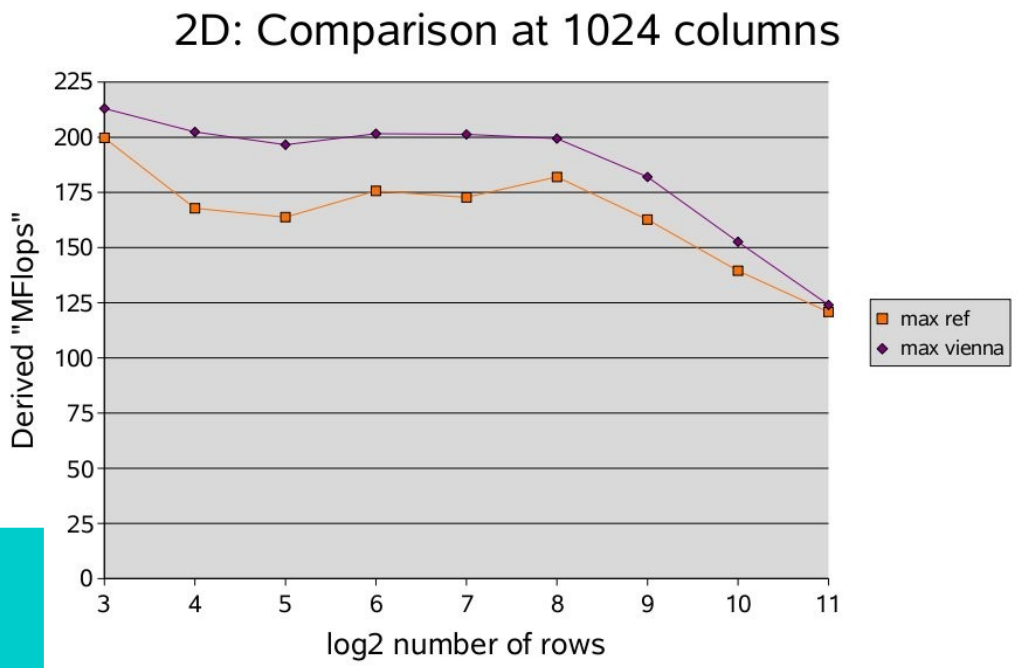
1K by 4K complex FFT shows >13% speedup over unmodified FFTW

Comparison of “Derived Mflops”, Forward Complex FFT



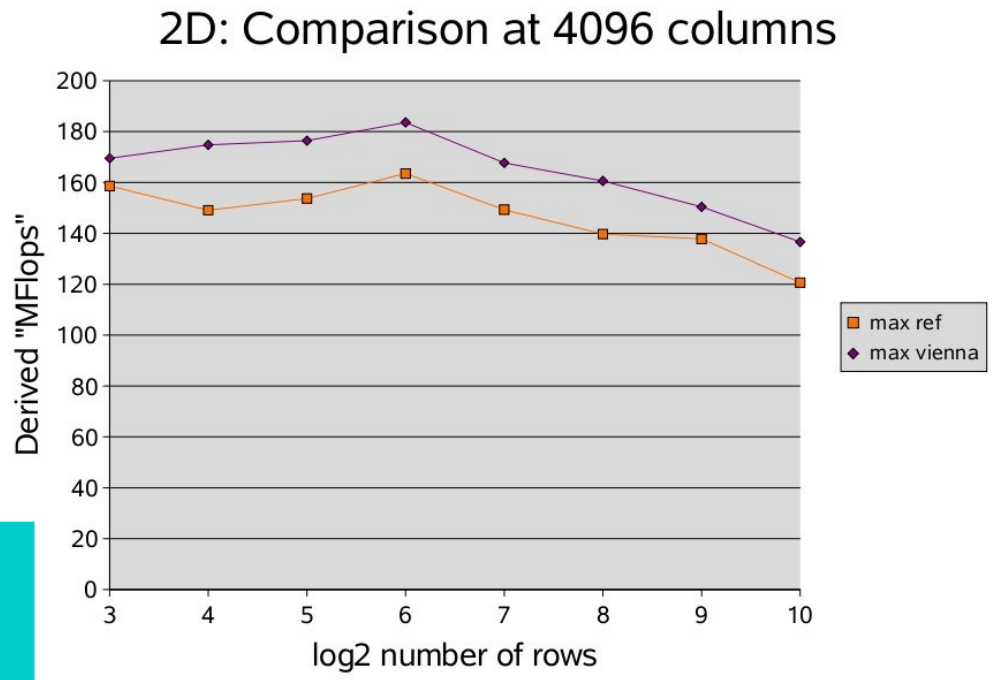
Scaled to a full size system the 1K benefit boosts performance by 0.5 TFlops.

Comparison of “Derived Mflops”, Forward Complex 2D FFT



Scaled to a full size system the 32x1024 tile benefit boosts performance by 0.25 TFlops.

Comparison of “Derived Mflops”, Forward Complex 2D FFT



Scaled to a full size system the 32x4096 tile benefit boosts performance by 180 GFlops.