

CONCLUSION*

Ivan Selesnick
C. Sidney Burrus

This work is produced by OpenStax-CNX and licensed under the
Creative Commons Attribution License 2.0[†]

1 Conclusion

We have found that by using the split nesting algorithm for circular convolution a new set of efficient prime length DFT modules that cover a wide variety of lengths can be developed. We have also exploited the structure in the split nesting algorithm to write a program that automatically generates compact readable code for convolution and prime length FFT programs.

The resulting code makes clear the organization and structure of the algorithm and clearly enumerates the disjoint convolutions into which the problem is decomposed. These independent convolutions can be executed in parallel and, moreover, the individual commands are of the form $I \otimes A \otimes I$ which can be executed as parallel/vector commands on appropriate computer architectures[3]. By recognizing also that the algorithms for different lengths share many of the same computational structures, the code we generate is made up of calls to a relatively small set of functions. Accordingly, the subroutines can be designed to specifically suit a given architecture.

The number of additions and multiplications incurred by the programs we have generated are the same as or are competitive with existing prime length FFT programs. We note that previously, prime length FFTs were made available for primes only up to 29. As in the original Winograd short convolution algorithms, the efficiency of the resulting prime p point DFT algorithm depends largely upon the factorability of $p - 1$. For example, if $p - 1$ is two times a prime, then an efficient p point DFT algorithm is more difficult to develop.

It should be noted too that the programs for convolution developed above are useful in the convolution of long integer sequences when exact results are needed. This is because all multiplicative constants in an n point integer convolution are integer multiples of $1/n$ and this division by n can be delayed until the last stage or can simply be omitted if a scaled version of the convolution is acceptable.

By developing a large library of prime point FFT programs we can extend the maximum length and the variety of lengths of a prime factor algorithm or a Winograd Fourier transform algorithm. Furthermore, because the approach taken in this paper gives a bilinear form, it can be incorporated into the dynamic programming technique for designing optimal composite length FFT algorithms [2]. The programs described in this paper can also be adapted to obtain discrete cosine transform (DCT) algorithms by simply permuting the input and output sequences [1].

*Version 1.2: Sep 15, 2009 10:04 pm -0500

[†]<http://creativecommons.org/licenses/by/2.0/>

References

- [1] M. T. Heideman. Computation of an odd-length dct from a real-valued dft of the same length. *IEEE Trans. Signal Proc.*, 40(1):54–59, January 1992.
- [2] H. W. Johnson and C. S. Burrus. The design of optimal dft algorithms using dynamic programming. *IEEE Trans. Acoust., Speech, Signal Proc.*, 31(2):378–387, April 1983.
- [3] R. Tolimieri, M. An, and C. Lu. *Algorithms for Discrete Fourier Transform and Convolution*. Springer-Verlag, 1989.