

# 要 旨

## FFT の ストリーム指向並列処理方式の研究

妻鳥 恵三

近年，様々なネットワーク上に流れるデータが急増している．これら大量のデータを高速に処理することがますます要求されている．マイクロプロセッサの分野では性能を向上させるために，一般的にマルチコアシステム上でのスレッドレベル並列処理が用いられる．しかし，スレッドの起動，停止，同期に伴うオーバーヘッドや，メモリウォールといった本質的な問題により理想的な高速化が望めない．この問題を解決するために，効果的にデータ並列アプリケーションを処理する，ストリーム指向並列処理方式が有望視されている．

本研究では，高速フーリエ変換 (FFT) にストリーム指向並列処理を適用することに焦点を当てている．まず，FFT に固有の並列なデータフローは，データ依存関係と，バタフライ演算の相対的な流量を表現することによって明示される．この表現はストリームフローグラフ (SFG) と呼ばれ，本研究では，color 識別子を利用可能なハードウェア資源のもと，有効な並列インスタンスを区別するために導入した．さらに，円滑なデータフローのために，FFT の中間データストリームにアクセスするためのメモリインタリーブアルゴリズムを導入した．そして，その実現可能性を検証するため FPGA に実装した．その結果，非メモリインタリーブ FFT 回路と比較して，1024 ポイントの FFT 回路のための処理時間が 49% 短縮され，追加的に必要な回路は 60% 必要であることが分かった．

キーワード ストリーム指向並列処理，ストリーム駆動アーキテクチャ，ストリームフローグラフ，FFT

# Abstract

## A Study on Stream-Oriented Parallel Processing Schema for FFT

Keizou Mendori

In recent years, the amount of data flowing via various networks has been drastically increasing. High-speed processing of such massive data is thus required more and more. In the microprocessor area, the thread-level parallel processing on multi-core systems is commonly employed to accelerate the performance. But, ideal speeding up cannot be expected due to essential issues such as memory wall and overheads of thread invocation, stop and synchronization. In order to solve this problem, the stream-oriented parallel processing scheme is promising to process data-parallel applications effectively.

This study focuses on applying the stream-oriented parallel processing to fast Fourier transform. Firstly, parallel dataflow inherent in FFT is specified by representing data dependencies and their relative flow rate among butterfly operations. This representation is called stream flow graph (SFG). In the SFG, "color" identifier is introduced to distinguish parallel instances activated under available hardware resource. In this study, a memory interleave algorithm to access intermediate data stream of FFT is introduced to smooth dataflow and it is implemented on FPGA for feasibility study. As a results, processing time for 1024-point FFT circuit is reduced % with additional circuit 60% compared to the non-interleaved FFT circuit.

**key words** Stream-Oriented Parallel Processing, Stream Driven Architecture, Stream Flow Graph, FFT