

FPGA 自動オフロード対象拡大の初期検討

An initial study of expanding the scope of FPGA automatic offloading

山登庸次
Yoji Yamato

NTT 株式会社 ネットワークサービスシステム研究所
Network Service Systems Labs., NTT, Inc.

1. はじめに

近年、ヘテロジニアスハードウェア利用が増えているが、活用の壁は高い。私は、配置環境に合わせ、既存のコード変換等を自動で行い適切に動作させる、環境適応ソフトウェアを提案してきた[1]-[9]。これまでの FPGA への自動化は、主に個々のループ文をオフロードするかどうかの検討だった。しかし、FPGA はハードウェア処理であるため、手動改造に比べ十分高速化できなかつた。本稿では、個々ループ文より大粒度計算の FPGA 自動オフロードを対象とする。

2. FPGA オフロード対象範囲拡大のアイデア

以前論文で提案した、ループ文の FPGA への自動オフロード方式では、算術強度やループ回数が高くリソース効率が高いループ文に絞って、オフロードパターンを作り、検証環境測定を通じ高速パターン探索を行うことができ、自動化を実現していた。しかし、個々ループ文に対しオフロードするか判定する方式では CPU のみに比べ数倍程度のある程度高速化はできて、極めて大きい高速化は難しかった。

なぜなら、FPGA はハードウェア処理の特性を生かし、パイプライン処理等を駆使して高速化することが多いため、画像処理、信号処理、行列計算、ステンシル計算等、計算タイプに応じて FPGA 処理のアルゴリズムから考える、手動での高速化が殆どだからである。そこで、個々のループ文に対し判定するのではなく、より大きな粒度の計算タイプに応じた計算処理に対し、多くの方が別論文等で今までに検討している FPGA 処理アルゴリズムを適用する事で、自動での高速化を行う。

動作概要としては、以下の 2 ステップからなる。まず、オフロードしたいコードに、FPGA オフロードできる計算タイプの計算処理が含まれるかを分析する。それが含まれている際に、その計算処理の FPGA 処理に該当する既存ノウハウが含まれた実装に置換することで処理を高速化する。ここで、FPGA オフロードできる計算タイプの計算処理が見つからない場合は、以前提案のループ文の FPGA オフロード高速化の試行に移行する。

検索のため事前に、FPGA にオフロードできる計算タイプ（画像処理、信号処理、行列計算、ステンシル計算等）のコード、その検索パターン、それを FPGA で処理する場合の OpenCL のコードを DB に保持しておく。この DB の情報は、FPGA オフロード高速化に用いられるので、FPGA インスタンスを提供するクラウド事業者が FPGA インスタンスの利用活性化を狙い準備する事を想定している。各計算タイプの計算処理を FPGA で処理する OpenCL に関

しては、別の方論文等で検討され実装された無料の物を用いる。

コードを分析し、計算タイプを判定するため、個々の変数名や関数名には依存しない抽象語を用いて意味的に、計算タイプに応じたプログラム構造に対し、マッチングできることが必要であると考ええる。

FPGA にオフロードできる計算タイプの計算処理を含むコードか判定できたら、検索された部分を FPGA で処理する場合の OpenCL のコードに置換することで高速化する。OpenCL のコードは、画像処理、信号処理、行列計算、ステンシル計算等で他論文等で手動改造で高速化が検討されてきた計算タイプであり、専門家の今までのノウハウが詰まった実装と言える。

3. まとめ

本稿では、私達が提案している環境適応ソフトウェアの拡張として、ユーザが提供するアプリケーションを、個々のループ文によらず分析し、画像処理、信号処理等の処理の計算タイプに応じて、適切な処理アルゴリズムで FPGA に自動オフロードする方式を提案した。今後は、一つは多くの実用アプリケーションに対して、計算タイプに応じた FPGA 自動オフロードを試行する。もう一つは、現在アクセラレータを用いたオフロードは FPGA、メニーコア CPU、GPU であるが、計算タイプに応じた自動オフロードを別ハードウェアにも適用する。

参考文献

- [1] Y. Yamato, "Proposal of Automatic GPU Offloading Method from Various Language Applications," ICIET 2021, pp.400-404, 2021.
- [2] Y. Yamato, "Study and Evaluation of Automatic Offloading Method in Mixed Offloading Destination Environment," Cogent Eng., Taylor and Francis, Vol.9, Issue.1, 2022.
- [3] Y. Yamato, "Proposal and evaluation of GPU offloading parts reconfiguration during applications operations for environment adaptation," J. Netw. Syst. Manag., Springer, 2023.
- [4] Y. Yamato, "Study and evaluation for adopting environmental adaptation of low-resource devices," IEEE Access, 2024.
- [5] Y. Yamato, "Study and Evaluation of FPGA Reconfiguration during Service Operation for Environment-Adaptive Software," Int. J. Parallel Emergent Distrib. Syst., Taylor and Francis, 2023.
- [6] Y. Yamato, "Study and Evaluation of Optimum Location Deployment for Environment Adaptive Applications," Int. J. Parallel Emergent Distrib. Syst., Taylor and Francis, 2022.
- [7] Y. Yamato, "Proposal and Evaluation of Adjusting Resource Amount for Automatically Offloaded Applications," Cogent Eng., Taylor and Francis, Vol.9, Issue.1, 2022.
- [8] Y. Yamato, "Study and Evaluation of Automatic Offloading for Function Blocks of Applications," Automatika, Taylor and Francis, Vol.65, Issue.1, pp.387-400, 2024.
- [9] Y. Yamato, "Study and evaluation of automatic division of general-purpose programs to facilitate addition of user functions," Int. J. Parallel Emergent Distrib. Syst., Taylor and Francis, 2024.