

環境適応ソフトウェアのオフロード対象拡大の初期検討

An initial study of environment-adaptive software offloading targets

山登庸次

Yoji Yamato yoji.yamato@ntt.com

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、多様な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]。これまでの多様なハードウェアへの自動化は、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、多様なハードウェアはハードウェア特性を生かした処理が多いため、手動改造に比べ十分高速化できなかった。本稿では、個々ループ文より大粒度計算の多様なハードウェアへの自動オフロードを対象とする。

2. 多様ハードウェアオフロード対象範囲拡大

以前論文で提案した、ループ文の多様なハードウェアへの自動オフロード方式では、オフロードする場合を 1、しない場合を 0 として遺伝子を模してオフロードパターンを作り、検証環境測定から高速パターン探索を行うことを遺伝的アルゴリズムを用いて行い、探索を自動化してきた。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少コア CPU での処理に比べ 10 倍程度のある程度高速化はできても、極めて大きい高速化は難しかった。

なぜなら、多様なハードウェアは多数コア等のハードウェアの特性を生かし、パイプライン処理等も駆使して高速化することが多いため、フリー変換等、計算タイプに応じて多様なハードウェア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、従来の個々のループ文に対しオフロードするしないを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、多くの他者が別論文等で今までに検討している多様なハードウェア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、多様なハードウェアに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、その計算処理の多様なハードウェア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、多様なハードウェアに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多様なハードウェアオフロード高速化の試行に移行する。

検索のため事前に、多様なハードウェアにオフロードできる計算タイプ（フリー変換等）のコード、その検索パターン、それを多様なハードウェアで処理する場合のコード（GPU なら OpenACC/CUDA, FPGA なら OpenCL, メモリーコア CP なら Open MP）を DB に保持しておく。この DB の情報は、多様なハードウェアオフロード高速化に用いられるので、多種の VM を提供するクラウド事業者が VM 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多様なハードウェアで処理するコードに関しては、他者の論文等で検討され実装された OSS のコードを用いる。

最初にコードを分析し、多様なハードウェアにオフロードできる計算タイプかを判定するため、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。

多様なハードウェア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、検索された部分を多様なハードウェアで処理する場合の実装に置換することでアプリケーションを高速化する。置換されるコードは、フリー変換等の他者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、フリー変換等の計算タイプに応じて、適切な処理アルゴリズムで多様なハードウェアに自動オフロードする方式を提案した。今後は、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多様なハードウェア自動オフロードを試行する。FPGA、メモリーコア CPU は別論文で詳細に評価しているので、全体像を検討する際は GPU を題材に評価する。

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, 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, 2022.
- [8] Y. Yamato, "Automatic Verification Technology of Software Patches for User Virtual Environments on IaaS Cloud," J. Cloud Comput., Springer, Vol.4, No.4, 2015.
- [9] Y. Yamato, et al., "Automatic GPU Offloading Technology for Open IoT Environment," IEEE Internet Things J., 2018.