C-Based Adaptive Stream Processing on Dynamically Reconfigurable Hardware: A Case Study on Window Join

Eric Shun Fukuda[1], Hideyuki Kawashima[2], Hiroaki Inoue[3], Taro Fujii[4], Koichiro Furuta[4], Tetsuya Asai[1], and Masato Motomura[1]

[1] Hokkaido University, Sapporo, Japan
[2] University of Tsukuba, Tsukuba, Japan
[3] NEC Corporation, Kawasaki, Japan
[4] Renesas Electronics Corporation, Kawasaki, Japan

Stream processing is becoming an important application field for reconfigurable architectures due to its two trends: 1) data rate and power consumption of a stream processing system is rapidly increasing, and 2) data streams have wide characteristics that should be dealt with different hardware architectures. However, utilizing reconfigurable processors requires hardware development skills which software engineers who develop algorithms do not have in general.

In our research, we consider window join, one of the operators of stream processing, as a case study and illustrate the current situation in two viewpoints: a) how a software engineer in this field can facilitate hardware acceleration utilizing a reconfigurable hardware with a C-level programming environment, and b) how adaptiveness is achieved in stream processing in such a solution. We use a dynamically reconfigurable architecture that features a state-of-the-art high level synthesis tool which compiles C source code into a hardware configuration.

We carried out a step-by-step optimization starting from a simple and ordinary software code. As a result, we achieved throughput improvement by two orders of magnitude compared to the first step, and roughly 50 times greater throughput/power efficiency than a pure software solution running on an Intel Core i5 CPU. We also evaluated two different architectures with each specialized in high/low match rate data streams. The architecture specialized in low match rate scored 45% higher throughput than the other when the match rate was 0.01%, while the latter scored 62% higher when the match rate was 10%.

We conclude that: i) a stream processing system with a dynamically reconfigurable hardware acceleration can be developed entirely in C, and achieve much higher throughput at a higher power efficiency, and ii) dynamically reconfigurable hardware is an efficient way to deal with streams with changing characteristics. However, there were still occasions that hardware design skills were required during the optimization steps. Therefore, we suggest that establishing a stream-oriented programming model which further hides hardware details and provides a relatively high performance without optimizing the code is necessary.

Acknowledgements. This work was partially supported by JSPS Grant-in-Aid for Challenging Exploratory Research (No. 24650033).