Design Space Exploration for Hardware Acceleration of Machine Learning Applications in MapReduce

Katayoun Neshatpour[1], Hosein Mohammadi Mokrani[1], Avesta Sasan[1], Hassan Ghasemzadeh[2], Setareh Rafatirad[1], and Houman Homayoun[1]

[1]George Mason University
[2]Washington State University

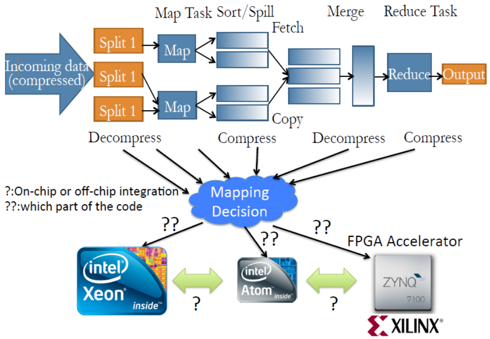

Fig. 1: Hadoop MapReduce Framework and the mapping decisions

***Abstract*—Emerging big data applications heavily rely on machine learning algorithms which are computationally intensive. To meet computational requirements, and power and scalability challenges, FPGA based Hardware accelerators have found their way in data centers and cloud infrastructures. Recent efforts on HW acceleration of big data mainly attempt to accelerate a particular application and deploy it on a specific architecture that fits well its performance and power requirements. Given the diversity of architectures and ML applications, the important research question is which architecture is better suited to meet the performance, power and energy-efficiency requirements of a diverse range of ML-based analytics applications. In this work, we answer this question by investigating how the type of FPGA (low-end vs. high-end), and its integration with the CPU (on-chip vs. off-chip) along with the choice of CPU (high performance big vs. low power little servers) affects the speedup yield and power reduction in a CPU+FPGA architecture for machine learning applications implemented in MapReduce. We show that among the three architectural parameters, the type of CPU is the most dominant factor in determining the execution time and power in a CPU+FPGA architecture for MapReduce applications. The integration technology and FPGA type comes next, with the power and performance least sensitive to the FPGA type.**

I. Experimental Setup and Methodology

Figure 1 shows the overview of the design space exploration of MapReduce in this paper. A total of eight machine learning algorithms including Kmeans, K nearest neighbor (KNN), singular value decomposition (SVD), support vector machine (SVM), hideen markov models (HMM), logistic regression (LR), collaborative filtering (CF) and Naive Bayes are implemented in Hadoop MapReduce.

For implementation purposes, we studied two very distinct server microarchitectures; a high performance big Xeon core and another a low power embedded-like little Atom core. These two types of servers represent two schools of thought on server architecture design: using big core like Xeon, which is a conventional approach to designing a high-performance server, and the Atom, which is a new trajectory in server design that advocates the use of a low-power core to address the dark silicon challenge facing servers.

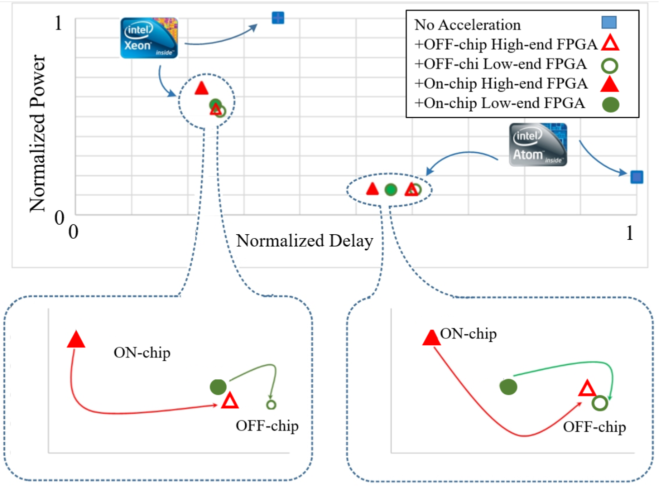

Fig. 2: Interplay effect of the parameters on power and delay.

For the choice of FPGA technology we studied two different generations of FPGAs, an Artix-7 representing a low-end FPGA and a Virtex-6 representing a high-end FPGA.

It should be noted that on-chip integration of the accelerator with the core allows faster transfer of data between the core and the accelerator. On the other hand, off-chip integration results in slower transfer rate between the two, but allows more flexibility, since we can choose the type of the FPGA and the core, without being confined to using the limited available on-chip FPGA+CPU platforms. To study the impact of integration technology, we model existing off-chip and on-chip interconnect protocols, PCI-Express Gen3 and AXI-interconnect, respectively.

II. Results Summary

Fig. 2 sums up the normalized results to better understand the importance of architectural parameters on both power and performance. As expected, Atom exhibits higher execution delays and lower power, while Xeon has lower execution time and higher power. We use the gap between various data points presented in the figure to find architectural insights. A major observation from Fig. 2 is that the type of CPU is the most important factor in determining the execution time and the power. The integration technology is the second important design parameter considering both power and execution time. The type of FPGA becomes of high importance only when the integration technology allows fast transfer of data between the CPU and the FPGA.