A Survey on Big Data Processing Infrastructure: Evolving Role of FPGA

Krishna Chaitanya Nunna[1], Farhad Mehdipour[2], Antoine Trouve[1], Kazuaki J. Murakami[1]

[1] Department of Advanced Informatics, Kyushu University, Fukuoka, Japan
E-mail: {krishna, trouve, murakami}@soc.ait.kyushu-u.ac.jp

[2] E-JUST Center, Kyushu University, Fukuoka, Japan
E-mail: farhad@ejust.kyushu-u.ac.jp

Abstract: In today's commercial world, information is becoming a major economic resource thus leading to a statement - Information is wealth. It is a technical challenge for computer systems in managing and analyzing the large volumes of data coming from a variety of resources continuously over a period. Experts are in a mood of moving towards alternative hardware platforms for achieving high-speed data processing and analysis especially for streaming applications. In this paper, (a) existing trends in big data processing and the necessary systems involved are studied by performing a survey on available platforms, (b) recommended features and suitable hardware systems are proposed based on the operations involved in the processing. Investigation shows that, in combination with CPU and along with GPU, FPGA is a possible alternative. It can be a part of the heterogeneous platform featuring parallelism, pipelining and high performance for the operations involved in big data processing.

Keywords: Big Data Processing; Data Analytics Infrastructure; 3D/Multi-FPGA System.

Reference to this paper should be made as follows: Nunna, K.C., Mehdipour, F., Trouvei, A., and Murakami, K.J. (2015) 'A Survey on Big Data Processing Infrastructure: Evolving Role of FPGA', *Int. J. Big Data Intelligence*, Vol. x, Nos. xx, pp. xxx–xxx.

Biographical notes:

Krishna Chaitanya Nunna received his master degree in 2007 from the School of Electrical Engineering of VIT University, Vellore (India). He later received his Dr. Eng. degree in 2014, from the Department of Informatics in Kyushu University, Fukuoka, Japan. He is a recipient of MEXT doctoral scholarship from Department of Education, Govt. of Japan for 2011-2014.

Farhad Mehdipour received his Ph.D. degree in Computer Systems Architecture from the Amirkabir University of Technology in 2006. He joined the School of Information Science and Electrical Engineering at the Kyushu University, Fukuoka, Japan as a post-doctoral researcher in December 2006. Since August 2010, he has been an associate professor in E-JUST Center at Kyushu University. He is a senior member of IEEE.

Antoine Trouvé received his Dr. Eng. degree in 2011 from the Department of Informatics of Kyushu University, Fukuoka, Japan. He has been working as a researcher at the Institute of Systems, Information Technologies and Nanotechnologies, Fukuoka until 2014. He is now working as an assistant professor at the Department of Advanced Information Technology of Kyushu University.

Kazuaki J. Murakami received the B.E., M.E., and Ph.D. degrees in computer science and engineering from Kyoto University, Japan in 1982, 1984, and 1994, respectively. From 1984 to 1987, he worked for the Fujitsu Limited, where he was a Computer Architect of the mainframe computers. In 1987, he joined the Department of Information Systems of Kyushu University, Japan. He is currently a Professor of the Department of Advanced Information Technology, and also the Vice President of the Institute of Systems, Information Technologies and Nanotechnologies (ISIT). He is a fellow of the IPSJ, and a member of the ACM, the IEEE, the IEEE Computer Society, the IEICE, and the JSIAM.

1 Introduction

In this globally-connected commercial world, information (or data) is becoming such key resource which is creating tremendous business opportunities and making the people keep saying "Information is Wealth". Now the data has reached in different directions in terms of size, type, and speed, and has received wide attention as "Big Data". It refers to the large amounts (volume) of heterogeneous data (variety) that flows continuously (velocity) within data-centric applications. All these are mentioned together as three Vs of big data (volume, variety, and velocity) (Russom, 2011) though not limited to three Vs. Volume, which is the primary characteristic of big data, refers to the large size (Tera or Petabytes) of records, transactions, tables, files, video, web text, sensor logs, and astronomical points, etc. Treating big data as big is because it is coming from the greater variety of sources that defines the second important characteristic as variety. Velocity refers to the frequency of data generation or the frequency of data delivery.

It is also necessary to consider two other equally important Vs: value and veracity. Value represents the analytic applications of the data and its potential associated value to the business. Veracity represents the quality and understandability of the data. That means that many users expect perfectly clean data. Putting all these Vs together, the commercial benefits of analyzing or mining such large set of data can be phenomenal especially in this so called social-connected global village. Study presented by various agencies such as MIT Sloan School of Management (LaValle et al., 2010), proved that the companies that use data analytics perform at least twice higher than the companies that don't use data analytics. To mention it, big data in companies is analyzed for many purposes such as: customer retention and approaching new customers at minimized cost, improving the future prospects of the analytics in the global market and many other commercial benefits.

Today's data centers are heterogeneous systems by combining Central Processing Unit (CPU) and Graphical Processing Unit (GPU) and sharing the workloads among each other based on the user requirements. Traditionally CPU and then GPU are the most popular machines for data management centers. Although each machine has its own benefits from the application perspective, it is proven (Fu et al., 2013) (Christos et al., 2012) (Che et al., 2008) that the GPU wins over CPU particularly for data processing. Field-Programmable Gate Arrays (FPGAs) are becoming as another choice of heterogeneous hardware due to their highly flexible, parallelism-oriented and reconfigurable architecture style. Choosing right platform among the available hardware based on the specific needs can be crucial in achieving necessary objectives. The decision of the necessary hardware depends on many factors and a good study on such space will provide enough benefits for the researchers to reduce the search path and reach the goal.

Here in this paper, the contributions include:

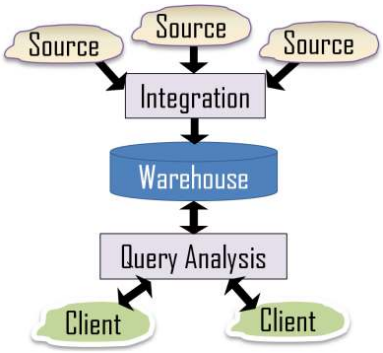
- Studying the available big data processing infrastructure and extracting the requirements of building processing platform.
- Recommending features and suitable hardware systems such as GPU and FPGA depending on various operations involved in big data processing. A comparison is carried out on GPU and FPGA by studying the available research works implementing various data operations.
- Investigating the role of FPGA in big data infrastructure using experimental analysis on FPGA based systems: multi-FPGA system and 3D FPGA.

In the following, Section 2 covers key stages of big data analytics, and Section introduces existing commercial and real-time processing infrastructure. The big-five features: Heterogeneity, Accessibility, Scalability, Protection, and Elasticity; necessary for big data infrastructure development are depicted in Section 4. Section 5 provides different processing systems in heterogeneous platform. Section 6 proposes systems selection based on the operations involved in big data processing, A comparison between GPU and FPGA through survey on available research work on both platforms is given in Section 7. Section 8 discusses FPGA's role in big data processing through experimental analysis and highlights FPGA's significance as a suitable candidate in big data processing platforms. Finally, the paper ends with the conclusion in Section 9.

2 Key Stages of Big Data Analytics

Systematically speaking, big data analytics is a technology involving the following key stages: Big data integration; Big data storage and processing; Big data query analysis and visualization; as shown in Fig. 1.

Figure 1 Key stages of big data analytics



2.1 Big Data Integration

Since the variety of data sources is extremely large in today's data management, linking and fusing different types of data is becoming a critical challenge. Within this challenge another major concern is not only integrating the different data types, but also dynamic behavior of data sources, quality, accuracy and timeliness of the data need to be considered. Dong et al. (2013) provide a neat discussion on state-of-the-art data integration techniques meeting the challenges of big data.

2.2 Big Data Storage and Processing

In an era of data analysis and management, storage is not only for data destination but also as a data platform. According to Redhat study (Redhat, 2013), five must-have fundamental requirements for big data management and storage are: cost-effective scalability, data migration elimination, bridging of disconnected discrete storage systems, data management through a unified data pool, and data availability and integrity through software.

Processing big data can be either in batch mode or streamline mode. That means some applications such as financial data are generated in batch mode. It is required to analyze and output the result on a scheduled basis, namely through the store-and-process paradigm. Many time-critical applications generate data continuously and expect the processed outcome on a real-time basis such as media processing involving video and image analysis. A new computing paradigm called Complex Event Processing (CEP) (Gulisano et al., 2012) deals with such issue. CEP generates complex events from a sequence of real-time events and allows events to be both filtered with user-defined patterns and transformed into new data. Hence, applications will be able to handle the events and data quickly and easily. Although CPU-based CEP systems achieve sophisticated event processing, they suffer from poor event processing performance. FPGAs are the possible reconfigurable hardware alternative in order to accelerate event processing (Inoue et al., 2011). Specific to batch processing, MapReduce (Apache, 2014) is a programming model and an associated implementation for processing and generating large data sets with a parallel and distributed approach. The hardware choice for such distinct processing needs must ensure the capability of handling and meeting the user requirements.

2.3 Big Data Query Analysis and Visualization

The enormous volumes of data require automated or semi-automated analysis that involves techniques to extract the credible information, to detect patterns or points, identify or match the objects among different images or videos. These kinds of techniques involve a combination of statistical analysis, optimization, and artificial intelligence along with new forms of computation. Innovative statistical models should be constructed to represent the unstructured data in a meaningful manner. The concepts such as machine learning play a pivotal role in automation of big data analysis. Once the data is mined, the outcome should be visualized according to the user requirement. For example, in the case of recommended systems, analysis algorithms should be intelligent enough to know which customer needs what (Zhen, 2013).

3 Existing Commercial and Real-time Big Data Infrastructure

Many technology firms developed their proprietary infrastructure for fulfilling the customer big data needs. Oracle explored typical use cases and proposed architecture decisions and necessary technology components which include variety of real-time applications such as; retail-weblog analysis, financial services real-time transaction detection; and insurance based cost-effective capturing of customers' driving habits and integrating with existing data (Oracle, 2012). Hewlett-Packard's HAVEn big data platform (Burk, 2013) is rapidly gaining its importance in its commercial big data analytics market expansion. HAVEn, is a big data analytics platform, which leverages HP's analytics software, hardware and services to create the next generation of big data analytics applications and solutions.

IBM's Netezza (Francisco, 2011), which falls under data warehouse appliance category, is widely credited for bringing renewed attention to the advanced analytics applications. They have developed a big data infrastructure platform using heterogeneous reconfigurable hardware such as FPGA. Their purpose-built analytics appliance includes custom-built FPGA accelerators. Netezza minimizes data movement by using innovative hardware acceleration. It uses FPGA to filter out extraneous data as early in the data stream as possible, and as fast as data can be streamed off the disk (Francisco, 2011). They proved and showed the tremendous benefits by introducing FPGAs in big data analytics hardware. Specifically saying, they compiled the queries using FPGAs to minimize overhead. Each FPGA on server blades contains embedded engines that perform filtering and transformation functions on the data stream. These engines are dynamically reconfigurable that enables them to be modified or extended through software. They are customized for every snippet through instructions provided during query execution and act on the data stream at extremely high speeds. Cisco Unified Computing System (Cisco UCS) introduced reliable scalability of hardware and management to increase

business agility, operational efficiency and helping in rapidly responding to changing business requirements (Cisco, 2014).

Academic researchers also showcased vital development in building infrastructure for big data analytics. For example, BlueDBM or Blue Database Machine, (Jun et al., 2014) is a storage system for big-data analytics that can dramatically speed up the time it takes to access information. In this system, each inbuilt flash device is connected to FPGA chip to create an individual node. FPGAs are used not only to control the flash device, but are also capable of performing processing operations on the data itself.

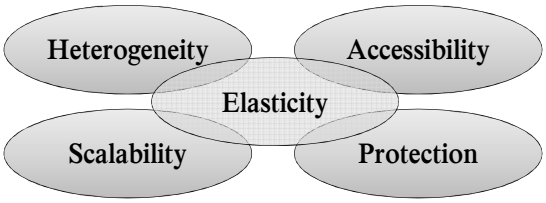
4 Big-Five Features Necessary for Big Data Processing Infrastructure

For the development of big data infrastructure, based on our study we have observed big-five features we refer as HASPE that need to be satisfied by the big data analytics infrastructure: Heterogeneity, Accessibility, Scalability, Protection and Elasticity (as shown in Fig. 2).

Heterogeneity: In the world of big data, the data sources responsible for such huge information are heterogeneous in the sense of data type. Data integration is responsible for handling such heterogeneity of input data. To improve the performance of processing big data, new hardware components are showing their tremendous features to be included in the overall system. For many decades, CPU has been the most popular and traditional system component for all needs. Introduction of innovations such as multi-core, many core processors, GPU and, to a certain extent, FPGA are used as accelerators in big data processing systems. One can think of other promising technologies such as systems having Massively Parallel Processor Array (MPPA) and large-scale reconfigurable data-path processor using single-flux quantum (SFQ) circuits (Mehdipour et al., 2011), which are specialized in accelerating scientific computations. These systems allow the designers to create heterogeneous hardware platform to improve performance and power efficiency. New approaches based on specialized heterogeneous processors such as FPGAs and general-purpose GPUs (GPGPUs) are being introduced into service with impressive results (Thomas et al., 2009).

Accessibility: The true objective of big data analysis is to create business opportunities. That means over many different kinds of applications; there will be as many users those are in need of specific and variety of data analytics outcome. To bring, add and satisfy all the users, big data infrastructure has to ensure that data users can access the data whenever and wherever they want. This means that it has to be both reachable and available from many systems across multiple locations.

Figure 2 Big-Five features for big data infrastructure



Scalability: Big data infrastructures are designated to handle large data volumes and certainly they need to be able to scale according to the user requirements over a period. The easiness in scaling the infrastructure decides the compatibility among different hardware and software components. Adding arrays of storage modules and/or improving the processing efficiency transparently without suffering from the overhead issue can add many advantages. They should be scalable geographically to enable the large infrastructures to be spread across multiple locations.

Protection: Technically saying, it is a combination of security and dependability. Majority of the applications carry their set of preferences for maintaining security requirements and standards. For example, US Federal agencies report a continuing shift to virtual desktop infrastructures for greater data centralization and deploying "cloud hubs"- a private-cloud infrastructure to maintain their own set of security standards for preventing data theft and hacking problems (Malykhina 2014). Citing necessary security standards specific to user requirements in the big data infrastructure development ensures the user for adequately classifying the risk level of data analytics and taking steps to mitigate risks.

Elasticity: Data sources and hence the data is getting larger and larger. Big data infrastructure designated to handle the present, and future data tends to be flexible enough in many ways. Care must be taken in the development so that they can grow and evolve along with the data sources. For example, so far the traditional data management requires centralized architecture components whereas the big data management required distributed architecture components for building efficient infrastructure. The overall system should be able to adopt technology trends necessary to cope with various objectives set by the applications. Normally scalability is related to the size of the infrastructure nothing but the storage capacity/expandable file space, and flexibility represents handling unknown requirements into the future. Elasticity is linked to all the above-listed features representing the much-needed storage, processing and analysis environment suited for fulfilling the demand generated by business requirements in omni-directions.

5 Processing Systems in a Heterogeneous Platform for Big Data Processing

Big data processing is per essence parallel regardless of the programming model. Parallelism can be achieved by executing CPUs in parallel, but it has been determined that is more power-efficient to have computation units on a single chip. Although modern CPUs do feature parallel units, there are other chips that propose more degree of parallelism such as GPU, FPGA, and MPPA. The former first two are already extensively available in the market and extensively studied in research. Although other processing systems like MPPA showed significant performance benefits, their usage is limited by many factors compared to GPU and FPGA (Thomas et al., 2009).

5.1 Graphics Processing Unit (GPU):

Computing paradigm witnessed transition from sequential computing, a dominant feature in the past, to parallel computing model. Such transition motivated the researchers for new and innovative computer architecture. GPU is perhaps the most successful new architecture. GPU is a highly specialized parallel processor for accelerating graphical computations. With the introduction of general computing on GPU with GPGPU mode, GPU has received wide applications ranging from the gaming industry to data analytics. That means GPU let the user perform flexible computation in a more general purpose sense. Several benefits can be attained using GPGPU: large performance through extended parallelism and cost-effective solution compared to CPU.

GPGPU is a combination of hardware components and software that allows the use of a traditional GPU to perform computing tasks that are extremely demanding in terms of processing speed. There are several popular systems exist in the commercial market from Nvidia, AMD (ATI), ARM Mali, and PowerVR. Also, many programming models are available for GPU such as CUDA (Nvidia, 2014), OpenCL, DirectCompute, C++AMP.

5.2 Field Programmable Gate Array (FPGA):

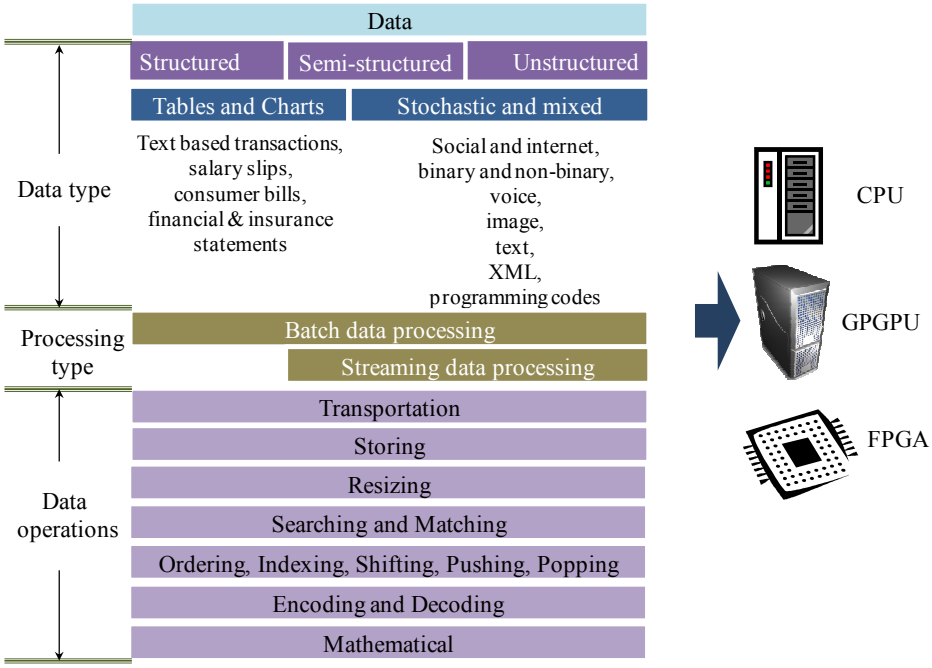
FPGAs can support very high rates of data throughput when high parallelism is utilized in circuits implemented in the reconfigurable fabric. FPGA reconfigurability offers a flexibility that makes them even superior to GPU for certain application domains. The purpose of an FPGA is to provide a customizable field-programmable device that can be optimized to perform the calculation for a specific problem. This is achieved by allowing the logic blocks on the chip to be logically re-connected even after the board has been shipped. The key features of FPGA that can provide motivation for big data analytics are: parallelism and efficient power consumption (performance/watt). Within FPGA technology, there are many different architectural implementations by different manufacturers in order to cope with the everyday changing technology trends such as multi-FPGA systems and 3D FPGA which will be discussed in later sections.

6 System Selection Based on the Data Operations

Most organizations with traditional data platforms such as enterprise data warehouses find that their existing infrastructure is either technically incapable or financially impractical for storing and analyzing big data. Companies such as Intel has developed and deployed a balanced platform for various real-world deployments of Hadoop (Apache, 2014), which is an open source distributed software platform for storing and processing data, runs on a cluster of industry-standard servers configured with direct-attached storage. Based on their assessments and benchmarking efforts, they have their recommendations for users or customers while considering infrastructure hardware. For example, Intel offers Xeon processor E5 for computing, essential system memory 48 GB to 96 GB of RAM per server, Intel SATA solid-state drives to fulfill the storage needs, and a minimum of 10 Gigabit Ethernet network (Intel, 2013). That means the selection of key specifications for a given application can be specific to each domain and operations involved in each system. Some of the most notable characteristics to help in decision making during the early stages of development are: bandwidth, data width, read/write speed, data process mechanism (e.g. batch and stream processing), network switching topology, traffic density, congestion control and security standards. All these characteristics define four distinctive categories of system specifications for hardware infrastructure including compute, memory, storage, and network.

Fig. 3 gives insight on the components derived from big data processing stage in Fig. 1. It shows different data types and data operations involved in big data processing. The big data comes from various sources such as social networking, mobile devices, satellites, financial items such as stocks, retail businesses, etc. All these data is structured, semi-structured or unstructured, often involve different data types such as tables, audio, video, text, html, etc. Based on the application domain needs, this data is required to be processed online (streaming) or offline (batch) which again requires storage and processing devices. The processed data is then analyzed specific to the user need and visualized in the necessary form. Even though this phenomenon of handling the data looks traditional in some way, the software and hardware architecture used for the entire flow varies heavily in order to fulfill big data needs.

Figure 3 Big Data Processing and Infrastructure Selection



Big data analytics discusses lots of relationships between various components, factors, sources and creates a platform to perform all the intended operations. For example, the core component of big data analytics - data mining, allows users to analyze data of many different dimensions, categorize it, and summarize the systematic relationships identified. Specifically, it is the process of finding correlations among heterogeneous fields in large databases.

Now the challenge is, what type of hardware system is more suitable for which data type? It is observable from our studies that, GPU, FPGA or CPU, or a combination of these components provides stupendous results in terms of performance and power efficiency. For example, GPU is power efficient but only for SIMD streams and FPGA is hard to program. However, the data flow style architecture feature in FPGA may dominate CPU and/or GPU in providing high-performance memory intensive operations at low power consumption (performance/watt) for a category of operations. Some of the typical operations involved in data mining stage in big data processing platform are Sparse matrix solution, Random number generation, Bayesian inference, Double precision floating point operations and so on. Several previous works showcased comparison of performing such vast number of operations on a different set of hardware systems. A study based on the type of hardware and a set of operations that can be efficiently implemented can be greatly helpful in choosing or recommending a suitable infrastructure for big data processing. Following section discusses effective contribution derived from our studies on earlier research works performing various operations on the intended hardware systems.

7 Performance and Power Efficiency in GPU and FPGA

In the past, many researchers showed the benefits of GPU and FPGA targeting data operations those can be encompassed in the real-time applications of big data. The following discussion gives the reader to compare GPU and FPGA for a wide range of operations covering various applications. Table 1 shows the list of technology domains/applications that can be involved in big data analytics. Each domain is studied based on a specific set of functions or operations which are necessary to be run or implemented on a selected hardware. We have studied previous research specific to each domain and a particular operation. Most of the works are targeted CPU, GPU and FPGA although some examined MPPAs. It is observed that the GPU and FPGA are the most competitive platforms for a vast range of applications.

Table 1 shows two columns indicating the superiority of the given platforms in terms of performance and power efficiency. That means; performance indicates how fast the given application can be run or implemented and how power efficient each platform is. For example, a 20x power efficiency for FPGA means, FPGA is 20 times more power efficient than its counterpart. In other words, FPGA consumes less power than its counterpart. The hardware chosen for these studies are available in the commercial market. It

should be noted that, it is practically difficult to select the different hardware resources featuring same characteristics because each one is unique in their respective segments. Especially the clock frequency selection is a major hurdle in comparing those systems. For example, typical FPGA will have less clock frequency than its counterpart. Sometimes its frequency is nowhere matching CPU/GPU. However for fair comparison, researchers made choosing the hardware based on a similar resource utilization by the operations. In most of the works, results are compared in terms of cycle counts, eliminating scaling and frequency issues. In some works, FPGA fabric's efficiency is evaluated relatively to the GPGPU by normalizing the operation's performance to device core count.

With respect to the applications in Table 1, each and every one is commercial and frequently used data sources generate large amounts of data. It is quite essential for the processing infrastructure to be compute-intensive for running and processing such variety of data under time-bounded format. Special-purpose processors such as accelerators are designed to speed up such compute-intensive sections of applications. GPU and FPGA are the possible accelerators which can often achieve higher performance than CPUs on certain jobs. Che et al. (2008) presented a comparison between GPUs and FPGAs by running three diverse operations-*Gaussian Elimination*, *Data Encryption Standard (DES*, and *Needleman-Wunsch* on both the systems and also on CPU. They have provided pros and cons of FPGA platform as they compare to GPU. Although the hardware characteristics of all three systems are not at the same level, which may not be possible to achieve, authors compared the results in terms of cycle counts thus avoiding scaling and frequency issues.

Table 1 List of observations for a variety of data sources processed on different systems.

Application Domain	Operations Involved	Reference Work	Performance		Power	
			GPU	FPGAs	GPU	FPGA
DNA Sequence Alignment	Gaussian Elimination, DES, Needleman-Wunsch	(Che et al., 2008)	12 x CPU (Gaussian)	50 x CPU (Gaussian) >1000 x GPU (DES), 15 x GPU	–	–
Astrophysics	Gravitational Calculations	(Hamada et al., 2009)	Relatively higher for old technology nodes	Better performance for the largest technology nodes	–	34 x CPU, 15 x GPU
Bioinformatics	Bayesian Interface Algorithm	(Fletcher et al., 2011)	-	3 x GPU	–	–
Climate Modeling, Geophysics Exploration, Remote Sensing	Parallel Data Compression, Parallel Sparse Matrics Solver	(Fu et al., 2013)	23 x GPU	330 x CPU 14 x GPU	14 x CPU	144 x CPU, 9 x GPU
Autonomous Navigation and Surveillance	Stereo Correspondence Algorithms	(Kalarot et al., 2010), (Ureña et al., 2012)	Significant internal overhead	Superior implementation	–	–
	Tone Mapping, Contrast Enhancement, and Glare Mitigation		More precision high-quality output images	Higher frame rates and less power	–	–
Molecular & Quantum Mechanics, Bioinformatics, and Fluid Mechanics	Random Number Generation	(Thomas et al., 2009), (Kestur et al., 2012), (Andryc et al., 2013), (Papakonstantinou et al., 2009), (Pratas et al., 2010)	9 x CPU	30 x GPU 3 x GPU	9 x CPU	175 x CPU, 18 x GPU
	Basic Linear Algebra Subroutines (BLAS), Double-Precision Floating-Point		~ CPU, > FPGA	~CPU but flexible, Reaching GPU	–	> CPU, > GPU
Financial Engineering Model	Heston Stochastic Volatility Mode	(Delivorias et al., 2012)	250 x CPU	590 x CPU	–	–

That means authors compared the values returned by performance counters via library functions. For the first application, FPGA showed superiority to other two systems. The major overhead of GPUs and CPUs comes from executing instructions that rely on memory accesses. FPGA took the advantage of data-flow streaming thus saving many of the memory accesses. In this application, the only drawback with FPGA is the programming complexity. For the second application DES, again, FPGA is superior to other two systems mainly because FPGA can finish bit-wise calculations in one cycle. Interestingly GPU does not support some important operations for DES whereas FPGA has no such problem. Note that it may also be possible to provide explicit support of bit-wise operations by software programming in GPU. In the last application execution, again FPGA achieved lowest overhead among all the three systems. But for a larger input size, the ratio of GPU execution cycles to FPGA execution cycles becomes smaller, due to better GPU utilization. Note that the data protection can be better achieved in FPGA than GPU due to its hardware programmability which is less vulnerable to hacking and counterfeiting.

Above application shows the comparison in terms of performance. Although, big data processing is a performance intensive, some applications specifically require reduced energy cost. Hamada (2009) presented research work on comparing FPGAs, GPU and General Purpose Processors (GPP) targeting many-body simulations for astronomical systems. They have compared all the systems in terms of development years, chip technology, pipeline depth, frequency, power consumption and similar other parameters. It is proved from their experimental results that FPGA could be a viable solution on an energy cost basis for very high performance, large scale many-body simulations.

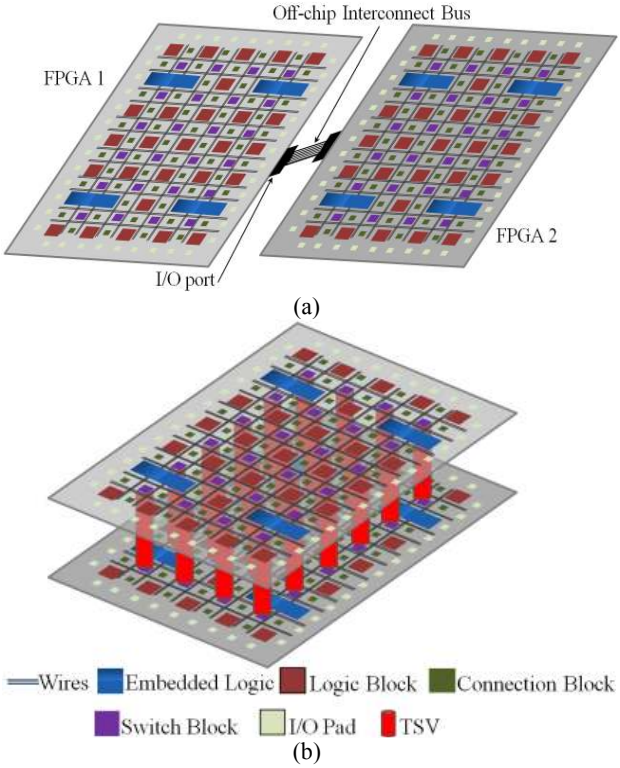
The work presented by Thomas et al. (2009) targets random number generation that is a frequently used function in high-performance computing (HPC). It can be efficiently implemented on FPGA than GPU/CPU as it is consuming 18 times less power than GPU and 175 times less power than CPU. FPGA wins for the majority of the applications compared to its counterparts in terms of power consumption (performance/watt). The major reason for this phenomenon is because of the easy memory access using the inbuilt memory banks. Furthermore, FPGA can finish bit-wise calculations in one cycle that can results in improved performance and reduced power consumption.

Applications such as climate modeling, geophysics exploration and remote sensing data processing require data compression and sparse matrix solver. Fu (2013) showed that FPGA outperforms GPU and CPU with at least 14 times greater performance (points/sec) than GPU and 330 times greater super-performance than CPU. Along with that, FPGA is nine times more power efficient (points/(sec x watt)) than GPU and 144 times than CPU, thus highlighting the FPGA's significance in such commercial applications.

Although FPGA is a winner with respect to the power-efficient category, still GPU is outperforming its counterparts in terms of performance for some commercial applications as shown in Table 1. Multimedia and communication algorithms from the HPC domain (Cullinan et al., 2012) often make extensive use of floating-point arithmetic operations. Due to the fact that complexity and expense of the floating-point hardware on a reconfigurable fabric such as FPGA are high, these algorithms are converted to fixed-point operations thus making FPGA less efficient than GPU for achieving higher speeds. Although such statement holds for many years, industry people are trying to mitigate this problem by developing floating-point data flow for streamlining the implementation process to enable those designs for achieving higher performance and efficiency as presented in Berkeley (2012). FPGA is still finding its importance in very specific set of data transfers (sending and receiving data) as reported by Cullinan et al. (2012).

It is worth mentioning that a combination of FPGA, GPU and CPU hardware infrastructure is giving good results for the applications such as medical imaging (Meng et al., 2012). Even though it tends to be very expensive to develop such true heterogeneous infrastructure, the choice is purely based on the user

Figure 4 (a) Multi-FPGA using off-chip interconnect bus (b) TSV-based 3D FPGA



requirement. Another significant requirement of heterogeneity is in signal processing domains which are in need of FFT, FIR, AES and floating-point operations. Several other works such as (Chase et al., 2008), (Fowers et al., 2012), (Grozea et al., 2010), (Haselman et al., 2012), (Jones et al., 2010), (Kapre et al., 2009), (Marrakchi et al., 2012), (Muthumala et al., 2012), (Nechma et al., 2012), (Pacholik et al., 2011), (Sarkar et al., 2010), (Yang et al., 2010) and (Zhang et al., 2009) have performed a comparison on above hardware platforms leading to similar conclusions.

8 FPGA Systems for Big Data Processing

One of the vital features of FPGA is its parallelism through hierarchical style architecture which can be very much suitable to data processing applications. Many of the widely used and typical data operations can be implemented on FPGA through hardware programmability. Researchers in the past showed many benefits attained by FPGA compared to CPU and GPU for a wide range of applications. Based on the applications, quantity and size of FPGA varies and has architectural constraint from the commercial device availability perspective. For the applications like big data processing, if someone wants to use FPGA as a part of the whole heterogeneous system, there is an immediate need of networking FPGA chips through off-chip interconnect buses. Such kinds of systems are already in use typically named as multi-FPGA systems. These are the devices developed to handle much larger designs compared to a single FPGA chip. Recently through the advancement of IC manufacturing, 3D integration of dies/wafers on to a single chip allows the designers to handle larger designs with better performance benefits through on-chip communication. The following sub-sections discuss these two different FPGA systems: multi-FPGA systems and 3D FPGA; and comparison among them in terms of area and performance efficiency through experimental approaches proposed in our previous work (Nunna, 2014) .

8.1 Multi-FPGA System

A multi-FPGA system, as shown in Fig. 4(a) contains multiple reprogrammable devices on a PCB. A system of FPGAs can be seen as a computing substrate with different properties than standard microprocessors. It provides a huge amount of fine-grain parallelism. When a circuit needs to be implemented on a multi-FPGA system, it is partitioned into a number of parts equal to the number of FPGA chips on the system. Then, these partitions are mapped onto those FPGAs separately. Inter-chip connections facilitate the communication between the FPGA chips. Note that the bus shown in Fig 4(a) is an example representation of such communication. In real-time, the way FPGAs connected depends totally on the type of chip package used. Even though multi-FPGA system can handle larger designs, due to their off-chip communication strategy the communication between the chips is limited by the bandwidth constraints imposed by the interface unit. With the constraint such as a limited number of I/O pads on FPGA, it is also necessary to multiplex the FPGA-to-FPGA signals, which further reduces the performance. One possible solution to achieve higher speed at the same level of circuit complexity is three-dimensional (3D) integration of FPGAs, introduced below.

8.2 3D FPGA

3D FPGA is one of the promising innovations which can provide benefits like increasing transistor density, reduced form factor, heterogeneous architectures and improvement in delay by significantly reducing the wire lengths of integrated circuits (Alexander et al. 1996). It is a multi-layer device stacked using through-silicon via (TSV) technology. That means the communication between the layers is done by using TSVs as shown in Fig. 4(b). The communication between the layers in 3D FPGA is on-chip, and hence it is quite obvious from the implementation perspective to expect higher speed compared to the off-chip communication platform.

Our previous work (Nunna, 2014) introduced an evaluation methodology for comparing a multi-FPGA system with TSV-based stacked 3D FPGA. Our study indicates an emphatic analysis on benefits attained by 3D FPGA against the multi-FPGA system while running complex designs or applications. According to our experiment results based on the standard benchmarks (VPR, 1997), the 3D FPGA is effective in reducing the wirelength and routing area by an average of 20.78% and 27.42% respectively compared to its 2D counterpart. A multi-FPGA system consisting of two FPGAs can have a footprint area larger than a 2-layer 3D FPGA plus additional off-chip interconnect bus area. In terms of performance, the 3D FPGA achieved a maximum of around 80% lesser delay compared to multi-FPGA system of two FPGAs (Fig. 5). These results provide the strong motivations for 3D FPGA to be considered as an alternative for processing platform in data management and analytics. Specific to big data analytics where speed is a high-level priority for many applications especially streaming data processing, 3D FPGA can provide improved delay characteristics. This kind of motivation can add extra potential to the already available feature - parallelism of FPGAs, which may results in much faster analysis of complex data.

8.3 FPGA as a Competitive Candidate

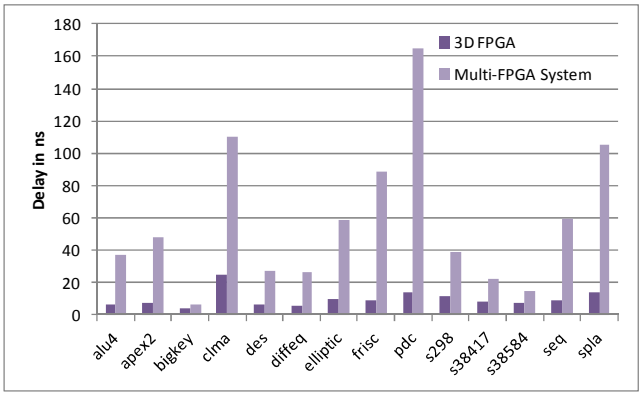
Over a range of application domains that continuously generate complex and unstructured data expect a highly efficient infrastructure for storage, processing and analysis needs. Traditional CPU may not be enough to fulfill and handle the extreme operational needs. Though multi-core and many-core architectures created a base for the last couple of years, GPU proved as the most replaceable candidate for CPU due to many advantages that we have discussed so far. GPUs are much cheaper than FPGAs. Software programming in GPUs is relatively easy compared to the hardware programming in FPGA thus making FPGA development difficult. It is understood that, although GPU gives optimistic parallelism by using its software programming concept, still FPGA can be a possible alternative to maintain the balance between power and performance which may not be a case in GPUs especially on an energy cost basis. The data flow and pipelining architecture style of FPGA gave an interesting choice to the designers to create a strong parallelism approach for data analytics. On the flip side, FPGA is finding its difficulty in floating-point operations and due to recent advancements and architecture development; it is becoming quite considerable across many applications. However, newer FPGA generations incorporate floating-point (FP) units as IP cores such as Stratix IV EP4SE530 (Altera, 2009).

FPGA can be considered as a processing engine in a cloud-based platform which requires a combination of distributed parallel processing and on-the-fly processing by demanding new technologies to fulfill the requirements such as high-speed data conversion, flexible resource allocation and resource optimization through load balancing (NEC, 2014). From the above discussions, we understand that for the applications, which need to make balance between performance, power, cost, time-to-market all together, FPGA can be an alternative along with GPU. Many companies such as National Instruments has already begun building FPGA-based virtual instrumentation such as reconfigurable IOs for commercial applications.

9 Conclusion

In this research paper, several developments in big data analytics were studied with specific concentration on hardware infrastructure. Big data processing trends were discussed by studying various data generation applications and different operations involved in some of the widely known applications. The survey was conducted based on the existing research works on some of the important data operations implemented on GPU, FPGA and CPU with respect to the performance and power consumption metrics. The significance of FPGA is pointed out as well from the comparisons. Within FPGA, different types of systems: multi-FPGA system and 3D FPGA were studied, and the experiment results showed that 3D FPGA wins in handling the objectives such as performance and area optimization.

Figure 5 Delay comparison: 3D FPGA vs. Multi-FPGA system



From our study we can say, CPU may not be enough for big data processing especially for future data-centric applications. It is observed that GPU and FPGA are relevant alternatives that are already in use. Although GPU is more common in use because it is easier to program and cheaper compared to FPGA, yet FPGA has the ability to occupy significant space in the heterogeneous platform of big data processing in many situations that we have studied. Also, to elevate the processing benefits of FPGA and supporting its architectural advantages, new FPGA technologies are coming to address its weaknesses.

References

Alexander, J. et al. (1996) 'Placement and Routing for Three-Dimensional FPGAs', *Proc. 4th Canadian Workshop Field-Programmable Devices*, pp. 11–18.

Altera Whitepaper (2009) 'Taking Advantage of Advances in FPGA Floating-Point IP Cores'.

Altera Whitepaper (2013), 'Radar Processing: FPGAs or GPUs? '.

Andryc, K. et al. (2013) 'FlexGrip: A Soft GPGPU for FPGAs', *Proc. Int. Conf. on Field Programmable Technology*.

Apache (2014) [online] http://hadoop.apache.org/

Berkeley Design Technology Report (2012) 'An Independent Analysis of Floating-point DSP Design Flow and Performance on Altera 28-nm FPGAs'.

Betz, V. and Rose, J. (1997) 'VPR: A New Packing, Placement and Routing Tool for FPGA Research', *Proc. Inter. Workshop Field Programmable Logic and Applications,* pp. 213–222.

Burke, S. (2013) [online] 'HP Haven Big Data Platform Is Gaining Partner Momentum', in CRN: http://www.crn.com/news/applications-os/240161649.

Chase, J. et al. (2008) 'Real-Time Optical Flow Calculations on FPGA and GPU Architectures: A Comparison Study', *Proc. 16th Inter. Symp. on Field-Programmable Custom Computing Machines*.

Che, S. et al. (2008) 'Accelerating Compute-Intensive Applications with GPUs and FPGAs, Application Specific Processors', *Proc. Symp. on Application-Specific Processors*.

Cisco whitepaper (2014) 'Cisco UCS and SAS: A Platform for Fast Data Analytics'.

Cullinan, C. et al. (2012) 'Computing Performance Benchmarks among CPU, GPU, and FPGA', in MathWorks.

Delivorias, C. (2012) '*Case Studies in Acceleration of Heston's Stochastic Volatility Financial Engineering Model: GPU, Cloud and FPGA Implementations*', MSc Thesis, The Univ. of Edinburgh.

Dong, X.L. and Srivastava, D. (2013) 'Big Data Integration', *Proc. 39th Int. Conf. on Very Large Data Bases*.

Fletcher, C.W. et al. (2011) 'Bridging the GPGPU-FPGA Efficiency Gap', *Proc. ACM/SIGDA Inter. Symp. on Field Programmable Gate Arrays (FPGA)*.

Fowers, J. et al. (2012) 'A Performance and Energy Comparison of FPGAs, GPUs, and Multicores for Sliding-Window Applications', *Proc. ACM/SIGDA Inter. Symp. on Field Programmable Gate Arrays*.

Francisco, P., (2011) 'The Netezza Data Appliance Architecture: A Platform for High Performance Data Warehousing and Analytics', in IBM Redbooks.

Fu, H. (2010) 'Accelerating Scientific Computing Through GPUs and FPGAs', Stanford Center for Computational Earth & Environmental Science (CEES).

Fu, H. (2013) 'Accelerating Atmospheric Simulation on GPU, FPGA, and MIC', Center for Earth System Science Tsinghua University.

Grozea, C. et al. (2010) 'FPGA vs. Multi-Core CPUs vs. GPUs: Hands-on Experience with a Sorting Application', *Lecture Notes in Computer Science: Facing the Multicore-Challenge*, Vol: 6310, pp. 105-117.

Gulisano, V. et al. (2012) [online] 'A Big Data Platform for Large Scale Event Processing', High Performance Computing on Graphics Processing Units: http://hgpu.org/?p=7484.

Hamada, T. et al. (2009) 'A Comparative Study on ASIC, FPGAs, GPUs and General Purpose Processors in the O(N^2) Gravitational N-body Simulation', *Proc. NASA/ESA Conference on Adaptive Hardware and Systems*.

Haselman, M. et al. (2012) 'FPGA vs. MPPA for Positron Emission Tomography Pulse Processing', *Proc. IEEE Inter. Conf. on Field-Programmable Technology*.

IBM Institute for Business Value Executive Report (2012) 'Analytics: The real-world use of big data'.

Inoue, H. et al. (2011) '20Gbps C-Based Complex Event Processing', *Proc. Intr. Conf. on Field Programmable Logic and Applications*.

Intel Big Data Analytics White Paper (2013) 'Extract, Transform and Load Big Data with Apache Hadoop'.

Jones, D. H. et al. (2010) 'GPU versus FPGA for High Productivity Computing', *Proc. Inter. Conf. Field-Programmable Logic Arrays.*

Jun, S.W. et al. (2014) 'Scalable Multi-Access Flash Store for Big Data Analytics', *Proc. Int. Conf. on Field-Programmable Gate Arrays*.

Kalarot, R. and Morris, J. (2010) 'Comparison of FPGA and GPU implementations of Real-time Stereo Vision', *Proc. IEEE Inter. Conf. on Computer Vision and Pattern Recognition Workshops*.

Kapre, N. and DeHon, A. (2009) 'Performance Comparison of Single-Precision SPICE Model-Evaluation on FPGA, GPU, Cell, and multi-core Processors', *Proc. Inter. Conf. on Field Programmable Logic and Applications.*

Kestur, S. et al. (2010) 'BLAS Comparison on FPGA, CPU and GPU', *Proc. IEEE Computer Society Symposium on VLSI*.

LaValle, S. L., et al. (2010) 'Big Data, Analytics and the Path From Insights to Value', in MIT Sloan Management Review.

Malykhina E. (2014) [online] 'Feds Look To Big Data On Security Questions', *Informationweek online magazine*.

Marrakchi, Z., et al. (2012) [onine] 'Improving ASIC Prototyping on Multiple FPGAs Through Better Partitioning', Tech Design Forum.

Mehdipour, F. et al. (2011) 'A Design Scheme for a Reconfigurable Accelerator Implemented by Single-Flux Quantum Circuits', *J. Syst. Architect Embedded Systems Design* 57(1), pp.169-179.

Meng, P. et al. (2012) 'FPGA-GPU-CPU Heterogenous Architecture for Real-time Cardiac Physiological Optical Mapping', *Proc. of International Conference on Field-Programmable Technology*.

Muthumala, W.H. et al. (2012) 'FPGA Implementation of Heterogeneous Multicore Platform with SIMD-MIMD Custom Accelerators', *Proc. IEEE Inter. Symp. on Circuits and Systems.*

NEC (2014) [online] 'Big Data Processing Platform', in Mobile World Congress.

Nechma T. (2012) '*Parallel Sparse Matrix Solution for Direct Circuit Simulation on a Multiple FPGA System*', Ph.D. Thesis, University of Southampton.

Nunna, K.C. et al. (2014) '3D FPGA versus Multiple FPGA System: Enhanced Parallelism in Smaller Area', *Proc. 12th Australasian Symposium on Parallel and Distributed Computing.*

Nvidia (2014) [online] http://www.nvidia.com/page/geforce_8800.html.

Oracle White Paper in Enterprise Architecture (2012) 'Oracle Information Architecture: An Architect's Guide to Big Data'.

Pacholik, A. et al. (2011) 'GPU vs. FPGA: Example Application on White-Light Interferometry', *in Proc. Inter. Conf. on Reconfigurable Computing and FPGAs*.

Papakonstantinou, A. et al. (2009) 'High-Performance CUDA Kernel Execution on FPGAs', *Proc. of 23th International Conference on Supercomputing*, pp.515-516.

Pratas, F. et al. (2010) 'Double-precision Floating-point Performance of Computational Devices: FPGAs, CPUs, and GPUs', *Proc. of the VI Jornadas sobre Sistemas Reconfiguráveis (REC)*.

Quan, J. (2013) 'The Implications from Benchmarking Three Big Data Systems', *Proc. IEEE Int. Conf. Big Data*.

Redhat Whitepaper (2013) 'The Five Must-Haves of Big Data Storage'.

Russom P. (2011) 'Executive Summary: Big Data Analytics', *in TDWI Research Best Practices Report*.

Sarkar, S. et al. (2010) 'Hardware Accelerators for Biocomputing: A Survey', *Proc. IEEE Inter. Sym. on Circuits and Systems*.

Thomas, D.B. et al. (2009) 'A Comparison of CPUs, GPUs, FPGAs, and Massively Parallel Processor Arrays for Random Number Generation', *Proc. ACM. Int. Symp. on Field-Programmable Gate Arrays*.

Ureña, R. et al. (2012) 'Real-Time Tone Mapping on GPU and FPGA', *EURASIP Journal on Image and Video Processing*.

VPR (1997) [online] 'VPR-Source Code and Benchmarks', http://www.eecg.toronto.edu/~vaughn/vpr/vpr.html.

Yang, D. et al. (2010) 'Performance Comparison of Cholesky Decomposition on GPUs and FPGAs', *Proc. Symp. on Application Accelerators in High Performance Computing*.

Zhang, Y. et al. (2009) 'FPGA vs. GPU for Sparse Matrix Vector Multiply', *Proc. Inter. Conf. on Field-Programmable Technology.*

Zhen, J. et al. (2013) 'Characterizing Data Analysis Workloads in Data Centers', *Proc. IEEE Int. Symp. On Workload Characterization*.