Call for Proposals

Computing Technology

HIRP OPEN 2017

Confidentiality

All information in this document (including, but not limited to interface protocols, parameters, flowchart and formula) is the confidential information of Huawei Technologies Co., Ltd and its affiliates. Any and all recipient shall keep this document in confidence with the same degree of care as used for its own confidential information and shall not publish or disclose wholly or in part to any other party without Huawei Technologies Co., Ltd's prior written consent.

Notice

Unless otherwise agreed by Huawei Technologies Co., Ltd, all the information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute the warranty of any kind, express or implied.

Distribution

Without the written consent of Huawei Technologies Co., Ltd, this document cannot be distributed except for the purpose of Huawei Innovation R&D Projects and within those who have participated in Huawei Innovation R&D Projects.

Application Deadline: 09:00 A.M., 16th June, 2017 (Beijing Standard Time, GMT+8).

If you have any questions or suggestions about HIRP OPEN 2017, please send Email (innovation@huawei.com). We will reply as soon as possible.

Catalog

HIRPO2017050301: Key technical progresses and challenges of superconducting quantum computing

1 Theme: Computing Technology

2 Subject: Superconducting Quantum Computing

3 Background

Quantum Computing is the art of controlling and exploiting the time evolution of highly complex, entangled quantum states of physical hardware registers for the purpose of computation and simulation. It has already been shown that difficult problems such as factorization of numbers can be efficiently solved using a quantum computer. Superconducting qubits use macroscopic circuits to process quantum information and are the most promising candidate towards this end. Indeed, the huge tech companies, such as Google, IBM, choose superconducting qubits to construct their quantum computers. Furthermore, some startup companies, such as D-wave, Rigetti, also prefer superconducting circuits. This project will be dedicated to acquiring knowledge and opinions on the state-of-the-art of superconducting quantum computing.

4 Scope

The project involves analyzing and integrating information about superconducting quantum computing in areas including, but not limited to, the following: basic principles, experimental apparatus and realization, different qubit implementations (Fluxonium, Transmon, Xmon, etc), computing architecture, scalability, circuit QED, quantum non-demolition measurement, fault-tolerant computing, quantum supremacy, commercial opportunities, current situation, and future trend.

5 Expected Outcome and Deliverables

Tutorial courses on the fundamentals and key technologies of superconducting quantum computing;

Survey reports about up-to-date progress of superconducting quantum computing;

1 top ranked conference paper or journal paper on superconducting quantum computing;

6 Acceptance Criteria

Tutorial courses/Survey reports/Conference paper to be reviewed and accepted by assigned acceptance team.

7 Phased Project Plan

Phase1 (~4 months): teach the tutorial courses about superconducting quantum computing, including basic ideas, key technologies, physical principle, etc.

Phase2 (~3 months): survey the state of the art of superconducting quantum computing, write the survey reports.

Phase3 (~2 months): write the paper about superconducting quantum computing.

Click here to back to the Top Page

HIRPO2017050302: Quantum algorithm overview, applications and potentials in ICT

1 Theme: Computing Technology

2 Subject: Quantum Algorithms

3 Background

Quantum computers are designed to outperform classical computers by running quantum algorithms, which can be applied to every fields, including cryptography, optimization, search, simulation of quantum systems, artificial intelligence and machine learning. The theory of quantum algorithms has been an active area of study for over 20 years. Indeed, the 'Quantum Algorithm Zoo' website cites 262 papers on quantum algorithm. This project will be dedicated to acquiring overview on the quantum algorithms, especially, crossover between machine learning and quantum algorithm, and the experimental feasibility of some special quantum algorithms like optimization.

4 Scope

The project involves analyzing and integrating information about quantum algorithm in areas including, but not limited to, the following: basic ideas about quantum algorithm, experimental feasibility, quantum algorithm examples, quantum supremacy, quantum machine learning, and future trend.

5 Expected Outcome and Deliverables

Survey reports about up-to-date progress of quantum algorithm;

Survey reports of quantum machine learning algorithm and experimental feasibility analysis;

6 Acceptance Criteria

Tutorial courses/Survey reports/Conference paper to be reviewed and accepted by assigned acceptance team.

7 Phased Project Plan

Phase1 (~4 months): survey the state of the art of quantum algorithm, including Shor algorithm, Grover algorithm, quantum simulation, quantum random walk, HHL's algorithm, etc.

Phase2 (~5 months): survey the state of the art of quantum machine learning and experimental feasibility, including optimization problem, using machine learning method to solve quantum physical problem, using quantum algorithm to speed up machine learning, etc.

Click here to back to the Top Page

HIRPO2017050303: Adaptive messaging mid-ware for distributed systems

1 Theme: Computing Technology

2 Subject: Design a Novel Messaging Scheme

List of Abbreviations

ROS – robot operating system

HMI – human –machine interface

GUI – graphical user interface

ML – machine learning

3 Background

A commonly used messaging scheme is known as ROS, which is an IP-network based, packet switching mechanism to handle communication demand over heterogeneously networked, multi-agent systems. The principle of ROS is illustrated as Fig. 1.

Fig. 1 Start and Runtime of ROS messaging

There're two type of nodes in ROS – master and slave nodes. A master node acts the conductor of the system as start-up, calling for event publisher

and subscribers to all other nodes. The slave nodes then will establish a network based on the master's advertisements and self interests. Once the topologic relationship is settled, the master agent will no longer be functional in system runtime, and slave nodes will send and receive subscribed events autonomously.

However, as ROS messaging framework is built upon IP networks and events are wrapped in UDP datagram or TCP streams, congestion emerges when traffic demand goes intensive, and consequently system performance degradation can be significant caused by packet latency and loss.

Therefore improvement can be achieved by re-designing the overall architecture of messaging mid-ware.

4 Scope

This project is aimed at feasibility study of a new messaging mid-ware for distributed systems that may comprise of different physical components and are used in different scenarios, typically, for domestic service. We summed the problems to be addressed in a new messaging mid-ware as follows-

1. Conserve traffic without compromising system performance
2. Learn and adapt the "true" traffic demand and priorities of the system in use phase, rather than merely relying on design

5 Expected Outcome and Deliverables

Documentation of problem analysis and solution, including but not limited to, performance evaluation report, technology approaches, and quantitative analysis, etc.

6 Acceptance Criteria

Report of feasibility study and long-term R&D proposal to implement a new messaging midware that may achieve higher performance, self-adaptation and ROS compatibility.

7 Phased Project Plan

Phase 0 (~3 months): delivery of technology survey and feasibility study.

Phase 1 (~6 months): delivery of evaluation report and conclusion.

Click here to back to the Top Page

HIRPO2017050304: Design AI chip architecture for training phase

1 Theme: Computing Technology

2 Subject: AI Accelerator

List of Abbreviations

AI: Artificial Intelligence

CNN: Convolutional Neural Network

RNN: Recurrent Neural Network

GPU: Graphics Processing Unit

3 Background

Progress in AI has been impressive this year. Those in the field acknowledge progress is accelerating year by year. In order to accelerate the AI computation, we need to design an AI accelerator to accelerate the AI algorithms.

AI algorithms involve two types of computing workloads with different profiles, known as training and inference.

In training, the network learns a new capability from existing data. Training is compute-intensive, requiring hardware that can process huge volumes of data.

In inference, the network applies its capabilities to new data, using its training to identify patterns and perform tasks.

Now there are many work about designing chips only for the inference phase, leaving the training phase to GPU. But GPU is not designed for AI in specialty.

So, it is a valuable research direction to design an AI chip architecture for training, to find a better way for training AI algorithms.

4 Scope

1) Research on characteristic of AI algorithms' training phase analysis: figure out the characteristic of different AI algorithms' training phase, algorithms should include the latest algorithms such as Neural Turing Machine, Generative Adversarial Networks, Reinforcement Learning etc.;

2) Research on AI chip architecture for training phase design: based on the analysis of AI algorithms' training phase, design AI chip architecture for training phase to accelerate AI algorithms' training.

5 Expected Outcome and Deliverables

Technical reports of analysis for characteristic of AI algorithms' training phase;

Technical reports of design for AI chip architecture for training phase, including theoretical analysis of the architecture, the performance simulation of the architecture design;

AI chip architecture with source codes and description;

1~2 Invention/patents and paper;

6 Acceptance Criteria

The proposed AI chip architecture can support the training for classical CNN (AlexNet/VGG/GoogLenet/ResNet)/RNN(LSTM/GRU) algorithms and the latest algorithms such as Neural Turing Machine, Generative Adversarial Networks, and Reinforcement Learning;

The proposed AI chip architecture have higher performance than GPU, have

lower energy than GPU;

7 Phased Project Plan

Phase1 (~3 months): survey the state of the art of algorithms in AI field, analyze the training characteristic of AI algorithms and provide the related technical report.

Phase2 (~5 months): research on AI chip architecture design for training phase, provide related design technical report and codes.

Phase3 (~4 months): research on AI chip architecture design simulation and verification, provide simulation results and patents and paper.

Click here to back to the Top Page

HIRPO2017050305: AI modeling platform support training and inference

1 Theme: Computing Technology

2 Subject: AI Chip Modeling

List of Abbreviations

AI: Artificial Intelligence

GAN: Generative Adversarial Networks

NTM: Neural Turing Machine

3 Background

Before develop the chip, we need to design the chip architecture, and we need the platform to explore the architecture, to evaluate the architecture, to find the issue and to optimize it. In the design of the AI chip, we need the AI modeling platform to model the AI chip architecture, to explore how to compute efficiency, to minimize the data movement, etc.

4 Scope

1) Research on AI Modeling platform in the industry and academe:

The state-of-the-art investigation report of the AI modeling platform that support training and inference in the academe and industry, and the analysis on these platform including advantages and disadvantages.

2) AI Chip modeling support AI Training and Inference:

Research on AI chip modeling platform that support the newest training network such as GAN and NTM. Flexibility and configurable to support the different chip architecture**,** and can evaluate the chip architecture and performance trending.

5 Expected Outcome and Deliverables

Technical reports of analysis the AI modeling platform in the academe and industry;

Technical report of the AI modeling platform for chip architecture, including training and inference and the newest network

AI modeling platform with source codes and description

1~2 Invention/patents and paper;

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.
Project deliverables are accepted by the evaluation team, Huawei.
Platform can support different training algorithm, support newest network such as Neural Turing Machine, Generative Adversarial Networks, and Reinforcement Learning.
The Platform can run at about 100Khz, support thread level parallel simulation.

7 Phased Project Plan

Phase1 (~3 months): survey the state-of-the-art of AI modeling in the academe and industry, analyzing the advantages and disadvantages.
Phase2 (~5 months): Research on AI modeling platform for training and inference provide the related design report.

Phase3 (~4 months): Research on AI modeling platform supporting the newest network, provide the related design and code and patents and paper.

Click here to back to the Top Page

HIRPO2017050306: Research on opengles/vulkan with software Implementation for android based on aarch64

1 Theme: Computing Technology

2 Subject: VMI (Software Graphic Render for Android)

3 Background

Opengles is standard for Embedded Accelerated 3D Graphics API,and Vulkan is a next-generation API from Khronos. Android N Integrate both Opengles and Vulkan As GPU API. Opengles and Vulkan creating a flexible and powerful low-level interface between software and graphics acceleration. Which provide high-speed graphics rendering for user. And Users can have high resolution and high FPS experience. And also can play Realistic 3D game on android system. Software Implementation for opengles/vulkan on aarch64 platform is a greater challenge,which is provide pure software graphics acceleration solution without hardware graphic card. Software graphics acceleration based on Aarch64 leads to new problem: how to Optimizing GLSL/ SPIR-V shader compiler. how to use NEON Instructions. And how to guarantee the Compatibility for massive Applications.

Shader is the most resource-intensive modules in modern graphics rendering system. And most shader compiler is design for hardware graphics acceleration. So Optimizing GLSL/ SPIR-V shader compiler for aarch64 platform is necessary, it include optimizing the memory access process, the data structure and floating point calculations.

NEON is a combined 64-bit and 128-bit SIMD instruction set that provides standardized acceleration for media and signal processing applications. It is very suitable for Vector operations, and the graphics rendering contains a lot of

vector calculation. So it can optimize graphics data structures and data access methods based NEON, for example. 3D Model include a large number of vertex and fragment data, how to send these data to cache and NEON register with low cache miss.

There are massive applications for android system. Different applications maybe dependent on different graphic api version on android. Such as opengles 2.0/opengles 3.0/vulkan. So we should make the software Implementation graphic api have good Compatibility.

4 Scope

The candidates is expected to deploy some deep research on opengles/vulkan with software Implementation for android based on aarch64 (but not limited to):

1) Propose method of GLSL/ SPIR-V shader compiler Optimization with NEON instructions based on aarch64 platform, based on mesa, swiftshader or other openGL|ES/vulkan software rendering lib.
2) Design a software opengles/vulkan prototype on android based on aarch64, which can decrease the time comsuming on software rendering processing to 30%~50%.
3) Provide 1~2 Patents and papers.

5 Expected Outcome and Deliverables

- Technical reports of Optimization with NEON instructions based on aarch64 platform GLSL/ SPIR-V shader compiler, including algorithms.
- A software opengles/vulkan prototype on android based on aarch64.
- Related simulation/evaluation platform with source codes and description.

- 1~2 Invention/patents, and 1~2 Publications in peer-reviewed Journals or top ranked conferences.

6 Acceptance Criteria

The proposed prototype can provide efficient 3D graphic acceleration.

The performance evaluation report about the prototype.

Improve the performance which can decrease the time comsuming on software rendering processing to 30%~50%.

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

7 Phased Project Plan

Phase1 (~3 months): survey the state of the art of 3D graphic accleartion, analyze and build the efficiency architecture and provide the related technical report.

Phase2 (~6 months): Building the prototype and the related technical report.

Phase3 (~3 months): Research and provide related evalution results, research publications and patents.

Click here to back to the Top Page

HIRPO2017050307: App cooperation in client-cloud system

1 Theme: Computing Technology

2 Subject: VMI (Client-Cloud)

3 Background

Cloud computing has tremendously changed the way of our live, work and study. Also, cloud service infrastructure has a huge influence on the way of business such as Google, Amazon and Microsoft. AWS avenue grows rapidly every year and has reach 12.22 billion U.S. dollars in the last year. With the benefits of Cloud services there are still some problems to be solved:

When we talk about the Cloud system is meaning the computing resource, large memory, big data with vast stroage system. This is the traditional Cloud system that working with enterprises. With the IoTs coming, the Cloud system will work for personal users, and the Cloud will cooperate with smart phone, smart watch, vehicles or other devices. These devices has a common feather: they don't have enormous resource eg. computing resource, memory or storage system. They like to share the resource in the cloud. But another problem is in the Cloud system, the architecture is usually CS/BS model application or distributed applications. This is a problem for mobile devices, we need some solution to solve the apps running on mobile devices interacting with cloud services.

4 Scope

The candidates is expected to deploy some deep research on client-cloud system, and it is suggested to focus on (but not limited to):

4) Propose research on Client-Cloud system in the industry and academe:

5) Provide a Client-Cloud system supports: apps in client system communicate and cooperate with server side Android cloud services, including apps, system services, file system and messages in VMs or containers.

6) Provide a Client-Cloud system supports: apps in cloud Android system communicate and cooperate with other Android cloud services including apps, system services, file system and messages in VMs or containers.

7) Provide a Client-Cloud system with apps in both client Android system and Android cloud system have no modifications, reprogramming or recompiling. All modifications are in the system architecture.

8) Provide Android cloud management in userspace and stateful Android cloud services.

5 Expected Outcome and Deliverables

- Client-Cloud System supports apps communicating and cooperating with each other in different side and with no modification in user-level.
- User management and stateful cloud services including:

  1) Android cloud service with user management

2) Stateful Android container

- The state-of-the-art investigation report of client-cloud system.
- Technical reports of client-cloud system and user management.
- 1~2 patents and paper

6 Acceptance Criteria

System can support Android apps communicating and cooperating with each other in different side (client/cloud) or node (cloud/cloud).

System can support Android cloud service user management and provide stateful cloud service.

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

7 Phased Project Plan

Phase1 (~3 months): survey the state of the art of client-cloud system, analyze and build the client-cloud system architecture and provide the related technical report.

Phase2 (~6 months): Research on client-cloud system, user management and stateful cloud services and building the prototype.

Phase3 (~3 months): Fulfill the related technical report, research publications and patents.

Click here to back to the Top Page

HIRPO2017050308: Intelligent energy efficiency resource management

1 Theme: Computing Technology

2 Subject: Intelligent Data Center Resource Management

3 Background

- In the Cloud Computing Era, hardware resources such as network, storage and computing are converged as different resource pools via virtualization technologies. Applications can be provided the needing resources easily. In fact, resource management is very important in data center operation system.
- Currently, the size of data center is much larger than before due to the heavy data processing requirements. The energy consumption is a big problem which will limit the size growing of data center.
- Currently many researchers are concentrated on reducing energy consumption of data center by using new energy efficiency device, employing renewable energy resources and introducing smart energy control system. However, few people take the energy efficiency resource management into their energy-aware data center research. It is valuable to investigate how to use new resource management to reduce the energy consumption of data center.

4 Scope

- **Research on Intelligent resource management system:** investigate and design the architectures and prototype of the energy efficiency resource management system, build the task or job scheduling theory model of the system;
- **Research on performance and energy consumption:** based on the intelligent system, design reasonable data center resource management model and build suitable scheduling theory model to increase the utilization rate of the and reduce the energy consumption, finally to get a lower PUE for the data center.

5 Expected Outcome and Deliverables

- Technical reports of architecture design and system analysis for Intelligent resource management, including resource granularity partition, scheduling algorithm analysis and benefit evaluation;
- Technical reports of energy analysis and performance analysis for Intelligent resource management, proving that the system can reduce energy consumption in the data center without performance consume;
- Platform prototype of Intelligent resource management with source codes and description;
- 1~2 Invention/patents;

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

The proposed resource management system can at least reduced energy consumption of data center by 10% without performance loss meanwhile.

7 Phased Project Plan

Phase1 (~3 months): survey the state of the intelligent resource management in industry and academic, and identify the problems, metrics and requirements in this topic, forms technical reports.

Phase2 (~5 months): Research on architecture and solution design, building a verifiable and measurable platform. Form the solution design report and the brief evaluation of the core idea.

Phase3 (~4 months): Research on performance and energy consumption of the platform, finding a suitable energy consumption ratio to support the 10% reduction of the energy consumption and without performance loss. And provide related algorithms, simulation results and patents.

Click here to back to the Top Page

HIRPO2017050309: Resource management and scheduling research for cloud-terminal fusion computing

1 Theme: Computing Technology

2 Subject: Cloud-Terminal Fusion Computing

3 Background

Currently, most of terminals have powerful computing abilities, such as mobile phone, car computer, game console and so on. All of these terminals connect with data centers which provide data computing, storage and complex task processing service. However, the size of data center grows bigger following the increment of terminals. How to reduce the computing pressure of data center and introduce the computing ability of terminal into computing processing is a difficult question. Cloud-Terminal Fusion is a trend of future computing architecture which can solve the question mentioned before. In this architecture, one computing task can be handled by terminal and data center together and the total processing time will be reduced.

So, it is a valuable research direction to analyze the resource management and scheduling of Cloud-terminal Fusion Computing.

4 Scope

1) Research on Cloud-Terminal Fusion Computing system: investigate the architectures and the use cases of the system, build the computing theory model of the system;

2) Research on resource management and scheduling: based on the computing model, design resource management approaches to choose the suitable resources between data center and terminal to fit the computing tasks. Design task scheduling algorithms to distributed different tasks on different places and reduce the whole processing time together with increasing the utilization of resources.

5 Expected Outcome and Deliverables

Technical reports of architecture and computing model analysis for Cloud-Terminal fusion computing.

Technical reports of resource management approach development and task scheduling algorithms design, including theoretical analysis, simulation results analysis and realistic demo.

Cloud-Terminal fusion computing resource management and scheduling simulation platform with source codes and description;

1~2 Invention/patents;

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.
Project deliverables are accepted by the evaluation team, Huawei.
Comparing with conventional cloud computing, total processing time is reduced by 15% under Cloud-Terminal fusion computing architecture with innovational resource management and scheduling.

7 Phased Project Plan

Phase1 (~3 months): survey the state of the art of Cloud-Terminal fusion computing architecture, analyze and build the computing model and provide

the related technical report.

Phase2 (~5 months): Research on resource management approach development design based on computing model to achieve the resource utilization increment and provide the related technical report.

Phase3 (~4 months): Research on task scheduling for Cloud-Terminal fusion computing and provide related algorithms, simulation results and patents.

Click here to back to the Top Page

HIRPO2017050310: Research on Fast R-CNN object detection architecture in embedded FPGA platform

1 Theme: Computing Technology

2 Subject: AI Accelerator

3 Background

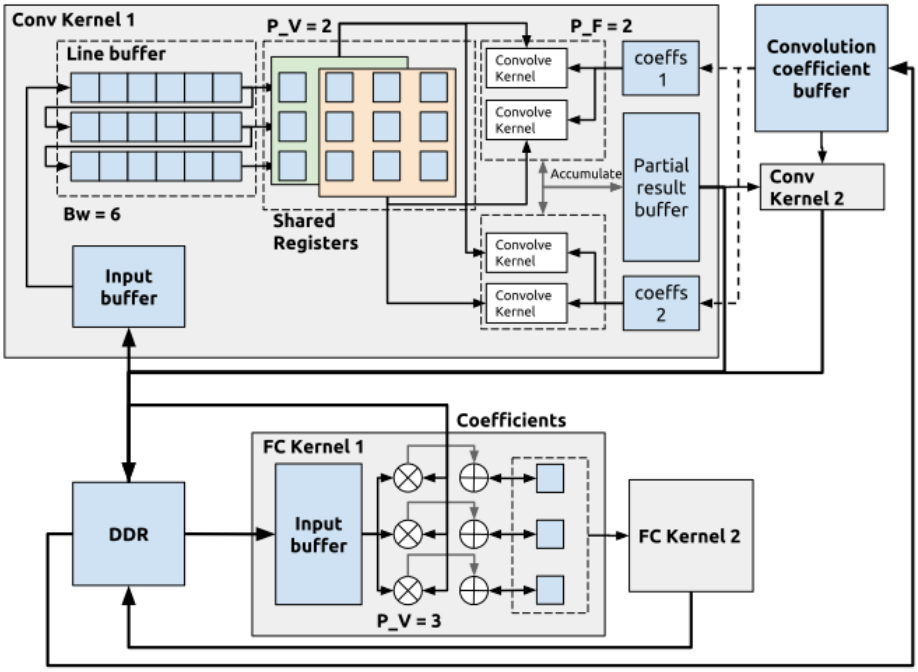
Microsoft proposes a Fast Region-based Convolutional Network method (Fast R-CNN) for object detection. Fast R-CNN builds on previous work to efficiently classify object proposals using deep convolutional networks. Compared to previous work, Fast R-CNN employs several innovations to improve training and testing speed while also increasing detection accuracy. Fast R-CNN trains the very deep VGG16 network 9× faster than R-CNN, is 213× faster at test-time, and achieves a higher mAP on PASCAL VOC 2012. Compared to SPPnet, Fast R-CNN trains VGG16 3× faster, tests 10× faster, and is more accurate. Fast R-CNN is implemented in Python and C++ (using Caffe) and is available under the open-source MIT License at https://github.com/rbgirshick/fast-rcnn.

Algorithms based on Convolutional Neural Network (CNN) have recently been applied to object detection applications, greatly improving their performance. However, many devices intended for these algorithms have limited computation resources and strict power consumption constraints, and are not suitable for algorithms designed for GPU workstations. The requirement need to presents a novel method to optimize CNN-based object detection algorithms targeting embedded FPGA platforms. Given parameterized CNN hardware modules, an optimization flow takes network architectures and resource constraints as input, and tunes hardware parameters with algorithm-specific information to explore the design space and achieve high performance. The evaluation need to be show model accuracy is above 85% and, with optimized configuration, our design can achieve 50 times speed-up compared with software implementation.

4 Scope

Research on Fast R-CNN Object detection architecture in embedded FPGA platform:

This section presents the basic architecture of our hardware design, which consists of two kernels: conv kernel and fc kernel. Each kernel contains an input buffer to cache data for further re-use, a computation kernel to perform convolution (conv) or matrix vector multiplication (fc), and an output buffer to store partial result before the final result is ready. Here we introduce these three components for each kernel in detail.



5 Expected Outcome and Deliverables

Design constraint VOC2012 dataset and Framework caffe.

1. Technical reports of Fast R-CNN Object detection architecture in embedded FPGA platform.
2. It is best to achieve Fast R-CNN in RTL Verilog code, which can be replaced by GPU realization for Job complexity.
3. Host Software stack carry on the real-time video detection.

The Deliverables can be discussed by communicate with each other.

1 Invention/patents;

6 Acceptance Criteria

Training use GPU, at Detection time use FPGA, features are extracted from each object proposal in each test image. Detection with Fast R-CNN takes latency per image on 10x faster than CPU) or 2x faster than Object detection based on VGG by GPU Realization

The Deliverables can be discussed by communicate with each other.

7 Phased Project Plan

Phase1 (~1 month): Research on Fast R-CNN Object detection architecture in embedded FPGA platform

Phase2 (~6 months): RTL Coding, which is best, or GPU coding

Phase2 (~3 months): host software coding and

Phase3 (~2 months): on –board testing.

Click here to back to the Top Page

HIRPO2017050311: Deep learning applications on ReRAM-crossbar

1 Theme: Computing Technology

2 Subject: ReRAM PIM Accelerator

List of Abbreviations

PIM: Process in Memory

ReRAM: Resistive Random Access Memory

3 Background

There are two ways to bridge the gap between computing and memory storage. First is to move more memory closer to CPU/GPU, HMC and HBM are two typical forms. Second is to move more computing operations into memory, Processing-in-memory (PIM) is a promising solution to address the “memory wall” challenges for future computer systems. In order to accelerate deep learning application, we need to design a PIM architecture which is different from traditional Von Neumann architecture.

An ideal nanoscale memristor crossbar array can naturally carry out vector-matrix multiplication in a single constant time step. By applying a vector of voltage signals to the rows of a memristor crossbar, multiplication by each memristor element’s conductance is carried out by the KCL rule and the current is summed across each column.

Since vector-matrix multiplication dominates the computation time and energy for neural network algorithms, it is a valuable research direction to design such

platform to accelerate deep learning application by utilizing the natural current accumulation feature of ReRAM-crossbar.

4 Scope

1) Research on ReRAM-crossbar specific deep learning neural networks: figure out the characteristic of deep learning neural networks and pick out the appropriate one which is suitable for ReRAM-crossbar implementation.

2) Research on ReRAM-crossbar platform: design large-scale ReRAM-crossbar platform, which consists of cascaded crossbars, explore the peripheral circuits for implement auxiliary functions (ReLU, pooling, normalization, etc) and cascade circuits to connect crossbars (ADC, DAC, etc) and how to map the network into the crossbar architecture.

5 Expected Outcome and Deliverables

Technical reports of analysis for characteristic of deep-learning neural networks (AlexNet/VGGNet/GoogleNet/ResNet, etc) and how to map them to the target ReRAM-crossbar architecture;

Technical reports of design for ReRAM-crossbar platform, including how to design peripheral circuits, how to connect two crossbars, theoretical analysis of the architecture, performance simulation of the architecture design;

ReRAM-crossbar platform with source codes and description (Simulator or Prototype);

1~2 Invention/patents and paper;

6 Acceptance Criteria

The proposed large-scale ReRAM-crossbar platform (at least two cascaded

crossbar) can support the implementation for medium or large size deep learning dataset (CIFAR-10/ImageNet), an end-to-end presentation demo should be given;

The proposed ReRAM-crossbar platform should have higher performance and lower energy than GPU;

7 Phased Project Plan

Phase1 (~3 months): survey the appropriate neural network which is suitable for ReRAM-crossbar, analyze their characteristic and provide the related technical report.

Phase2 (~5 months): research on ReRAM-crossbar platform, provide related design technical report and codes.

Phase3 (~4 months): research on ReRAM-crossbar platform simulation and verification, provide simulation and implementation results and patents and paper.

Click here to back to the Top Page

HIRPO2017050312: Boost data-intensive processing in the datacenter with accelerators based memory interface

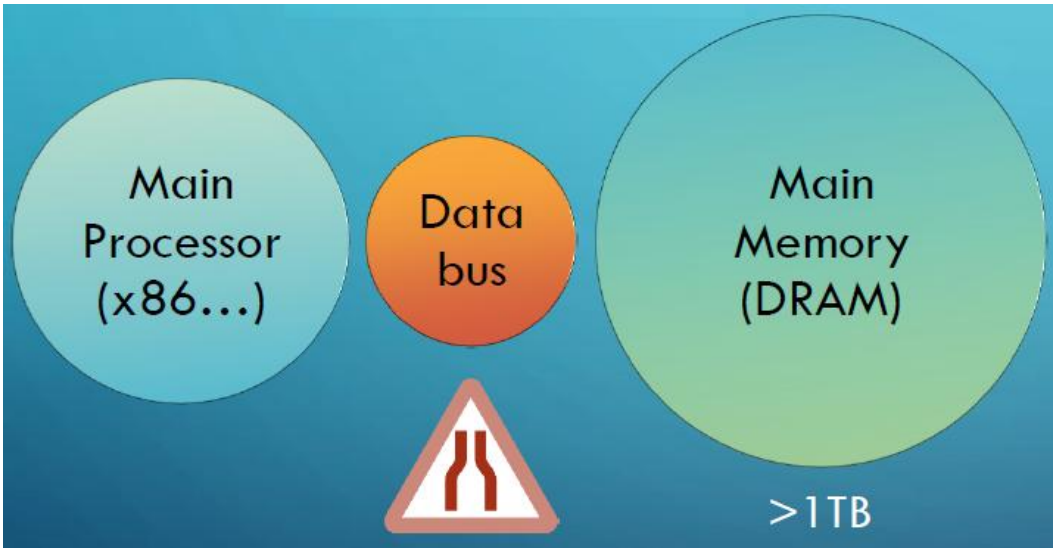
1 Theme: Computing Technology

2 Subject: Accelerator, Memory Technology

3 Background

Processing in large-scale systems is shifting from the traditional computing-centric model successfully used for many decades into one that is more data centric. Applications in the data center are processing more and more data. E.g. DB, Analytics, genomics, AI etc. Accelerating big data applications is key for the datacenter users.

However currently there are two bottlenecks: one is end of Moore's law for the server's processors, the transistor size is reaching a few atoms thick, and both integration and frequency are reaching their limits. The way is clearing for new solutions: heterogeneous architectures (FPGA, GPGPU, Co-proc.) are being implemented in the data center application servers. Another is hitting the memory wall, memory size is reaching 1 TB, and memory bandwidth is now the key limiting factor for performance.



So we need to find an approach to boost data-intensive processing in the datacenter with accelerators (processing in memory), meanwhile without disrupting the eco-system, e.g. still using memory interface and no impact on server architecture.

4 Scope

3) Research on Accelerator (PIM) in the industry and academe:

The state-of-the-art investigation report of the PIM accelerator architecture in the academe and industry, and the analysis on these accelerator including advantages and disadvantages.

4) Innovative processors arrays into the DRAM:

Research on innovative accelerator architecture, add many true co-processors within the main memory chips (DRAM) and let them process the data intensive operations much faster, under the host server CPU control, at the main cost of porting the core calculation software. Without any change in the server architecture. Finally give the performance evaluation on some benchmarks using simulation to deliver better memory performances than others.

5 Expected Outcome and Deliverables

Technical reports of analysis the PIM accelerator architecture in the academe and industry;

Technical report of the PIM accelerator's micro architecture, integrating innovative processing units within DRAM enables massive data processing power without the limits of bandwidth and latency that the host CPU. And give the performance evaluation how fast to accelerate data intensive applications on some benchmarks.

Cycle accurate simulator with source codes and description.

1~2 Invention/patents and paper;

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

The benefit is reasonable theoretically, and proved by the simulation evaluation; The PIM accelerator using memory interface and no impact on server architecture can deliver better memory performances than others (FPGA/GPU etc.), with performance evaluation on some benchmarks using simulation.

7 Phased Project Plan

Phase1 (~3 months): Survey the state-of-the-art of PIM accelerator architecture in the academe and industry, analyzing the advantages and disadvantages.

Phase2 (~5 months): Research on innovative accelerator architecture, add many true co-processors within the main memory chips, and provide the

related simulator design and patent.

Phase3 (~4 months): Performance evaluation on PIM accelerator with FPGA/GPU with simulation, provide the related design and code and patent and paper.

Click here to back to the Top Page

HIRPO2017050313: SCM based resilience in big data processing and neural network

1 Theme: Computing Technology

2 Subject: Computing Resilience

List of Abbreviations

SCM	Storage Class Memory
NN	Neural Network

3 Background

The Emerging new NVMs (like PCM, MRAM and ReRAM etc, also called SCM) bring the chance and challenge in computing and storage system. And currently, the big data processing and AI becomes the hottest applications in Industry. Spark is one of the most popular big data processing framework, it uses lineage - logging the transformations used to build a dataset, and using those logs to rebuild datasets when needed to keep application resilience, Additionally as an external storage system Tachyon handles redundancy and resiliency by, in a manner similar to Spark itself, keeping track of data computation "lineage" and re-computing results in case of failure.

SCM is very fast and can be accessed with CPU Load/Store directly, at the same time it can keep data persistent. So when Spark or Tachyon uses SCM, it can recover the data from SCM quickly and no need to use re-computing to recovery the result data.

How to use SCM as main memory, and at the same time to keep data consistency and failure recovery is a chance and challenge in big data processing and deep learning applications, and it will bring better performance, we believe.

So, it is a valuable research direction to investigate SCM based resilience in big data processing or AI applications.

4 Scope

SCM based resilience

Investigate how to combine SCM as main memory or storage in big data processing and AI frameworks, and using SCM to keep data consistency and provide fast failure recovery mechanism.

5 Expected Outcome and Deliverables

- Technical reports of SCM based Resilience in big data processing or AI applications;
- Related simulation platform with source codes and description
- 1~2 Invention/patents;

6 Acceptance Criteria

Design competitive SCM based Resilience in big data processing or AI frameworks, like Spark and TensorFlow.

7 Phased Project Plan

Phase1 (~3 months): survey the state of SCM based resilience and how to combine with Spark or TensorFlow;

Phase2 (~6 months): research on the data model in Spark or TensorFlow and how to store the data model in SCM and keep data consistency and can provide the fast recovery mechanism when failure.

Phase3 (~3 months): research and provide related algorithms, simulation results and patents.

Click here to back to the Top Page

HIRPO2017050314: NVML based new programming model and compiler auxiliary optimization

1 Theme: Computing Technology

2 Subject: Design NVML based New Programming Library and Compiler Auxiliary Optimization

List of Abbreviations

SCM	Storage Class Memory
PM	Persistent Memory
NVMP	Non Volatile Memory Programming
NVML	Non Volatile Memory Programming Library

3 Background

Programming NVM systems is the next major challenge.

NVML is an open-source library that simplifies development of applications utilizing byte-addressable persistent memory (PM). The SNIA NVM Programming Model describes basic behavior for a persistent memory-aware file system enabling applications to directly access persistent memory. NVML extends the SNIA programming model providing application APIs that help applications create and update data structures in persistent memory avoiding pitfalls such as persistent memory leaks and inconsistencies due to unexpected hardware or software restarts.

Based on the NVMP and NVML technology, we believe that new programming model, like NVML and Atlas, are the trend for using NVM. And some of them

relies on language extensions and compiler support, while others not. In this direction, how to construct a better programming model is valuable.

4 Scope

NVML based programming model

Investigate how to better use NVML, and improve efficiency, through optimizations which is NVM friendly, are integrated into NVML.

5 Expected Outcome and Deliverables

- Technical reports of NVM Based Programming Model, and compiling auxiliary support;
- Related simulation platform with source codes and benchmark.
- 1~2 Invention/patents;

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.
Project deliverables are accepted by the evaluation team, Huawei.
Design competitive NVML Based solution and source code.

7 Phased Project Plan

Phase1 (~3 months): survey the state of NVM Programming model and corresponding language extensions and compiler support.

Phase2 (~6 months): research on the optimization, like data structure, transaction, data flow mechanics, which is NVM friendly. Then integrate it into new Programming Model, like NVML.

Phase3 (~3 months): research and provide related algorithms, simulation results and patents.

Click here to back to the Top Page

HIRPO2017050315: Software scheduling framework and data model based on fusion of big data and neural network

1 Theme: Computing Technology

2 Subject: Scheduling Framework and Data Model

List of Abbreviations

SCM	Storage Class Memory
NN	Neural Network
NVM	Nonvolatile Memory

3 Background

The NVM has the good feature of speed closely to DRAM and persistence. It can improve the performance of Graphx at 15 time from some open source project, which only use local NVM. So,It is important to building a distributed pool of scm through RDMA/MPI/PGAS to speed up the applications of big data and neural network.

As we known, the GPU and special neural network library based on FPGA play the key role in the applications of big data and AI. These accelerate unit also can be pooled through communication technique similar with GPUDirect. And all the accelerate unit can be form a cluster such as google does.

So, it is a valuable research direction, to build a distributed platform to integrate the SCM pool and the accelerate unit, and use the advanced schedule algorithm to deploy the user data or training data to NVM pool and the program segment or AI algorithm to properly accelerate unit. In order to implement these ideas, Spark is good prototype, and maybe the spark core or shuffle will be improved.

4 Scope

SCM Pooled based on RDMA/MPI/PGAS

Investigate on the different collective communication(alltoall) and memory pinned(firehouse algorithm) technique in gasnet/MPI, and find out a design of building the SCM pool based on RDMA or improvement of MPI/Pgas.

Accelerate unit communication technique similar with GPUDirect.

Investigate on the GPUDirect technique and find out a bypass cpu design similar with “RDMA” between accelerate unit. And find out a design of building a cluster using GPU or accelerator.

Scheduling framework and data model for big data and neural network.

Investigate on different distributed AI framework and data model and layout, such as tensorflow. Investigate on scheduling framework or algorithm existing in spark, or latest Group Scheduling. Find out a fusion design of integrating the AI framework and dealing of big data in the same platform.

5 Expected Outcome and Deliverables

- Technical reports and design of scm pool based on rdma/mpi/pgas.
- Technical reports and design of communication library similar with GPUDirect between the accelerate unit.

- Technical reports and design of scheduling framework and data model for big data and neural network.
- Related simulation platform with source codes and description
- 1~2 Invention/patents;

6 Acceptance Criteria

Design competitive design of SCM pool, communication library between the accelerate unit and scheduling framework.

7 Phased Project Plan

Phase1 (~3 months): survey the art of building a distributed pool of SCM through RDMA/MPI/PGAS and the communication library similar with GPUDirect between the accelerate unit.

Phase2 (~6 months): research on scheduling framework or algorithm for the big data and neural network based on the SCM pool and different accelerate unit.

Phase3 (~3 months): provide a solution or research report and provide related algorithms, simulation results and patents.

Click here to back to the Top Page

HIRPO2017050316: Binary instrumentation on ARM v8

1 Theme: Computing Technology

2 Subject: Binary Instrumentation for Performance Tuning

List of Abbreviations

DBI: Dynamic Binary Instrumentation

GDB: GNU Project debugger

3 Background

We often insert some debug functions into application and Linux kernel to debug and tuning performance. Most of us modify the source code applications to achieve it. But it's very hard to maintain the codes with lots of debugging functions. For opensource codes we have to analyze them for weeks before we decide where we should do instrumentation. So we need a agile way to solve it.

4 Scope

1) Research on how to insert/delete user defined functions from running application and Linux kernel: figure out which method is better and fit for us and verify it (1. DBI like INTEL pin tool; 2. Exception control like GDB; 3. Hot patch fix)

2) Research on how to find out where the probe function should be insert: We can place the probe functions and find out the software/hardware bottleneck automatically.

5 Expected Outcome and Deliverables

1.A prototype tool (related source codes and specification) which can

a) Insert/delete user defined functions into/from both Linux user applications and kernel dynamically/statically.

b) Decide the positions where the functions should be inserted.

c) Find out the system bottleneck module/sourcce code/function via the functions automatically.

2.One~two Invention/patents

3.One paper accepted by CCF B

6 Acceptance Criteria

1.The prototype tool can run on new version CentOS(ARMv8)
2.Find out the software/hardware bottleneck by binary instrumentation tool automatically
3.Test workloads include speccpu int and nginx proxy
4.Overhead is less than 2%
5.One~two Invention/patents
6.One paper accepted by CCF B

7 Phased Project Plan

Phase1

(~2 months): investigate the best way to do instrumentation on ARM v8

(~4 months): develop a prototype tool can do instrumentation and a patent

Phase2

(~6 months): research on how to place the probe function and analyze the performance automatically and a patent

Click here to back to the Top Page

HIRPO2017050401: Evaluation of high performance computing device for deep learning algorithms

1 Theme: Computing Technology

2 Subject: Deep learning

3 Background

Deep learning algorithms have already been widely applied in various applications in Artificial Intelligence, while the significant computational complexity and the size of network parameters remain a challenge for current computing hardware devices. For example, in image classification and speech recognition applications, the deep learning algorithms usually contain tens of thousands of neurons and millions of parameters.

At the same time, new emerging hardware with high computing capability can help to accelerate the evolution of deep learning algorithm since the more powerful the hardware device is, the faster the training and inference of deep learning algorithms will be.

Currently, both large companies like NVIDIA, Google and startups such as Cambricon, Horizon Robotics are in the field of high performance computing device for deep learning algorithms. Therefore, it is necessary to have a professional, comprehensive evaluation of the performance of different hardware devices.

4 Scope

Targeting at existing high performance computing devices for deep learning algorithms, test the performance of the devices in different application

scenarios including image classification, object detection, speech recognition, etc., in different aspects and give comparative analysis in form of a evaluation report.

Evaluation requirement:

Hardware devices should include but not limited to:

HuaWei Hisi Hi** chip

NVidia P4, P40, TX1, TX2, TITAN Pascal

Other hardware platform such as Movidius, Cambricon, Intel, etc.

Application scenarios should include but not limited to:

Image classification, with different input size, uses different networks, with different number of classes for comparison

Object detection, with different input size, uses different networks, with different classes of objects for comparison

Speech recognition, with different input size, uses different networks, with different types of recognition for comparison

Performance evaluation results should at least include theoretical analysis and application result analysis:

Theoretical analysis should include but not limited to:

Computation complexity of inference

Number of parameters of inference

Accuracy of inference

Device computational capability

Application result analysis should include but not limited to:

Execution time of inference

Accuracy of inference

Power consumption of inference

Utilization rate of device

Software platform for testing should include but not limited to:

Caffe

TensorFlow

5 Expected Outcome and Deliverables

Configurations of evaluation should include but not limited to:

Detail configuration of hardware and software

Introduction of the deep learning algorithm

Evaluation result should include but not limited to:

Comparative analysis between the theoretical result and application result for each evaluation term

Comparative analysis among all terms used in evaluation

Source code for evaluation

6 Acceptance Criteria

Fail: No patents /report are delivered.

Pass: 1 patents pass Huawei's review AND, 1 detailed technical evaluation report AND corresponding source code

Excellent: More than 1 patents are delivered.

7 Phased Project Plan

Phase1 (~2 months): Survey top deep learning algorithms of Image classification, Object detection and Speech recognition. Deliverable includes: A survey of algorithms, Theoretical Analysis Report of Algorithms, Final Evaluation Report 1.0.

Phase2 (~2 months): Complete the evaluation environment setup for all the combinations among 2 kinds of software platforms and 4 kinds of hardware platforms. Complete the software and hardware support analysis evaluation report. Deliverable includes: Evaluation report on the support for different

combinations of software and hardware platform, Manual of evaluation environment configuration.

Phase3 (~3 months): Complete the evaluation of networks in different application field with varying input size, networks and number of classes. Complete the evaluation for different hardware platforms in terms of computation time, accuracy, power consumption and hardware utilization rate. Deliverable includes: Evaluation report on high performance computing devices for deep learning, Source code for evaluation, Raw data related to the evaluation.

Click here to back to the Top Page

HIRPO2017050501: Model building based on source code for problem location

1 Theme: Computing Technology

2 Subject: system model building and check

Project name: Model building based on source code for problem location

List of Abbreviations

NA

3 Background

Fjeldstad and Hamlen report pointed out: system performance improvement and error correction tasks, respectively, 42% and 62% of the time spent on understanding activities. In our complex embedded software development environment is the case, the system fault, the most commonly trouble shooting is log-analysis. Through the observation of the field log, the developer combined with the source code for the logical inference and analysis, analysis all possible paths, and eliminate the impossible path, to find out the root cause of the system fault. These analyses rely on manual labor, which is time-consuming and laborious. About Tens of millions lines source code, there is only theoretical feasibility for manual analysis. In the current cloud open scene, a large number of open source code bring more greater challenge, To maintain a huge system is very necessary to understand how the components of collaborative.

4 Scope

1) Investigate the auto problem location techniques in the industry and academia, based on source code and system running information, such as log, alarm, event, etc.

2) Research on how to build the system model based the system's static and dynamitic information. The model is used to represent the normal flow path and abnormal flow path and the dependence of these paths which is composed of system's objects and object's logical dependence.

3) Make analysis based on the upper system model. Given the real system's event or log, the software can analyze and output following info: the system execution path or the path is illegal (based on the user defined policy)

5 Expected Outcome and Deliverables

- Technical reports of auto problem location techniques based on the system's static and dynamic information;
- The prototype system and the design documents
  - ✧ **a system implementation model building software**

    Input:

    the system's source code and the running information;

    support user specified starting and ending points;

    support user specified header file;

    Output:

    the system implementation model which is be shown through graphics.

    Support the distributed multi-instances system.
  - ✧ **analysis system based on upper model**

    Input:

the target system real information (conclude log, event, alarm, etc).

Output:

the target system's fact execution path;

judge is there an exception.

- 1 patent
- 1-2 papers

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

The time of model analysis is less than 10s for 100M bytes log information, samples from Huawei.

The model accuracy rate is greater than 90%, samples from Huawei.

Language C/C++

7 Phased Project Plan

Phase No.	Phase description	Time(months)		Main task content	Output Standard that should achieve
1	Industry trend and prototype design	T	T+3	Survey on industry trend, key technologies manufacturer and market application, make the failure type and policy-making dimensions complete list; make the prototype design from this; make the list of test cases(benchmark).	Technical investigation report, design document and test cases.
2	Prototype implementation and emulation	T+3	T+7	Code, emulation result report.	None

3	Performance optimization	T+7	T+9	Code, emulation result report.	None
4	Acceptance	T+9	T+11	Final prototype design document, prototype code and test report）	Design documents, prototype code, comparison test report

Click here to back to the Top Page

HIRPO2017050502: Fast failure detection technology for cloud-distributed cluster

1 Theme: Computing Technology

2 Subject: Fast failure detection for cloud-distributed cluster

List of Abbreviations

FD	Failure Detection
CT	Communications Technology
IT	Information Technology
ICT	Information Communications Technology

3 Background

Under the scenario of cloud distribution, the problems such as the node crashing or packet loss, the link failure or packet loss, the network partition, and cluster failure resulted by disk failure, etc are always major difficulties in the distributed field. From CT to ICT, it is inevitable to meet these problems, and it is more sensitive and the requirements are stricter than those of IT. In the IT field, high tolerance is given to these problems mentioned before as well as resource (such as memory, cpu) consumptions. However, in the CT filed, the requirement for communication is very high and the resources are

expensive, as a result we have to ensure the cluster reliability and low resource consumption. For CT, high reliability and low resource consumption are guaranteed by dedicated hardware. From CT to ICT, the common hardware replaces the dedicated hardware, and it needs to guarantee the high reliability and low resource consumption by using software technology. A lot of products have the same problem, but there has not found a good idea to solve this problem in the industry. We really need a fast failure detection method for cloud-distributed cluster, which can quickly and accurately detect the fault, including nodes crashing or packet loss, link failure or packet loss and network partition, etc. It can greatly shorten the fault detection time, and improve the accuracy, which in turn minimize the loss of the service.

Ps: The “node” is a logical concept, it may be a process, a virtual machine, a docker or a physical node.

4 Scope

1) Investigate the fast failure detection techniques in the industry and academia, which focus on problems such as cluster nodes crashing or packet loss, the link failure or packet loss, the network partition, cluster failure resulted by disk failure and so on.
2) Research on how to quickly detect the fault with low resource consumption.
3) Research on how to build a cloud-distributed cluster failure model.
4) Make analysis of the failure based on the cluster failure model to improve accuracy.

5 Expected Outcome and Deliverables

- Technical reports of fast failure detection techniques for cloud-distributed cluster;
- The prototype system and the design documents

- ✧ a fast failure detection software
  Input: the node address information of a cluster (including scale in and scale out);
  Output: the failure detection result and relevant information
- ✧ a failure model construction and analysis system
  Input: failure information
  Output: failure model and failure type

- 1 patent
- 1-2 papers

6 Acceptance Criteria

Project proposal is accepted by the evaluation team, Huawei.

Project deliverables are accepted by the evaluation team, Huawei.

The time of failure detection is less than 100ms or real time(challenging goals: 10ms), samples from Huawei.

CPU usage rate is less than 1%, samples from Huawei.

Memory usage is less than 5M, samples from Huawei.

Faulire detection accuracy is larger than 99%, samples from Huawei.

Language C/C++

7 Phased Project Plan

Phase No.	Phase description	Time(months)		Main task content	Output Standard that should achieve
1	Industry trend and prototype design	T	T+3	Survey on industry trend, key technologies manufacturer and market application, make the failure type and policy-making dimensions complete list; make the prototype design from this; make the list of test cases(benchmark).	Technical investigation report, design document and test cases.
2	Prototype	T+3	T+7	Code, emulation result	None

	implementation and emulation			report.	
3	Performance optimization	T+7	T+9	Code, emulation result report.	None
4	Acceptance	T+9	T+11	Final prototype design document, prototype code and test report）	Design documents, prototype code, comparison test report

Click here to back to the Top Page

HIRPO2017050503: Acceleration of Graph Analytics based on processor-FPGA system

1 Theme: Computing Technology

2 Subject: Heterogeneous computing

List of Abbreviations

FPGA	Field Programmable Gate Arrays

3 Background

There is increasing interest in using Field Programmable Gate Arrays (FPGAs) to accelerate computation, particularly for big data processing on cloud platforms and in data centers. This interest is driven by the growing need for high-performance energy-efficient computing. Today, several vendors have systems that integrate FPGAs into data-center platforms, including ones from IBM, Intel, Xilinx and Microsoft. It is safe to predict that this interest will continue to grow over the next few years and play a central role in the design and use of cloud and data center computing platforms.

One class of FPGA-accelerated systems has recently emerged as a dominant platform for data center acceleration: systems that tightly integrate the FPGA with the processor to share system memory. These systems enable an accelerator circuit implemented on the FPGA fabric to directly read from and write to memory in manner that is coherent with the processor’s caches. A key advantage of this form of integration of is that both processor threads and an FPGA circuit can concurrently perform a computation, sharing data at a fine granularity. This, in turn, can result in higher performance gains and may enable new classes of applications to benefit from FPGA acceleration.

However, there are challenges to exploiting FPGAs for the acceleration of big data applications on these platforms. First, it is unclear the extent to which big data applications can benefit from concurrent processor-FPGA acceleration, given the diverse, and in some cases the irregular, nature of these applications. Second, it is also unclear how to best present FPGA accelerators to programmers. Big data applications are expressed using frameworks such as Hadoop or Spark. Thus, an important concern is how to integrate FPGA accelerators into these frameworks in a way that preserves their APIs, which programmers are already familiar with. Finally, FPGAs are programmed with hardware design abstractions and the associated tools can take excessively long times. This makes the use of FPGAs foreign and tedious to most software and application developers.

The overall goal of this project research is to tackle the above challenges. More specifically, the project aims to explore the benefits and potential overheads of the simultaneous use of processor threads and FPGA circuits to accelerate big data applications, and determine the extent to which such tight integration of the processor and the FPGA can open up opportunities for acceleration of big data applications. Further, this project tries to explore how to best integrate the FPGA accelerator into common data center frameworks.

4 Scope

This proposal primarily focus on following task 4.1: Acceleration of Graph Analytics based on processor-FPGA system(HIRPO2017050503)

This project consists of the following three specific aspects,

4.1 Acceleration of Graph Analytics based on processor-FPGA system(HIRPO2017050503)

The project will the benefits of tight processor-FPGA integration in the acceleration of big-data graph analytics. In particularly, by focusing on a tightly-coupled processor-FPGA system, (1) it will explore the concurrent use of processor threads and the FPGA accelerator; (2) it will use the shared memory space to alleviate limitations arising from data copying; and (3) it will focus on more application-oriented graph analytics problems.

A FPGA Accelerated Function Units (AFUs) will be designed and implemented to speed up the applications. CPU threads and these AFUs will be used simultaneously, through fine-grain sharing of graph data, to assess (1) the ability of an AFU to accelerate applications with such irregular data access patterns and (2) the benefits of using both processor threads and the AFU to improve performance.

4.2 Integrating FPGA Accelerators in Spark by Compiler technology for Heterogeneity(HIRPO2017050504)

4.2.1 Integrating FPGA Accelerators in Spark

The project will exploit the presence of shared memory between the processor and the FPGA accelerator to achieve better and more transparent integration. The goal is to present the accelerator to the software so it appears as “faster” thread of execution to the JVM runtime, able to fully access data in shared memory. It will be achieved through three mechanisms. In this way, the Spark runtime can appropriately manage the FPGA accelerator, allowing better utilization and load balancing.

4.2.2 Compiling for Heterogeneity

Compiler technology will be developed to ease the use of an FPGA-based heterogeneous system, also in the context of big data processing. Specifically, it will be conducted in two main directions. The first is developing compiler analyses that aid application programmers in determining which segments of

their code are most amenable to FPGA acceleration. These analyses consider not only code characteristics, but also the capabilities of the underlying overlays. The second direction also utilizes overlays to allow the generation of an accelerator circuit for code segments on-the-fly, as threads are scheduled for execution on a target machine. This enables the same code base to execute on either a CPU-only machine or on an FPGA-enabled machine with no explicit modifications/actions by the programmer.

5 Expected Outcome and Deliverables

- **Publications:** 3~5 research publication submissions are expected to document the novelty and results of our research, ~2 publications on the graph analytics component, ~2 publications on integrating FPGAs into the Spark framework, and ~1 publication on the compiling for heterogeneity.
- **Software components:** new LLVM-based compiler passes for application analysis, which will be released to Huawei upon their completion in the course of the project.
- **Accelerated function units**: the designs of these hardware components (in the form of Verilog files and associated documentations) will be also released to Huawei upon their completion in the course of the project.
- **Presentations:** periodical presentations to Huawei to ensure the expected progress and results.
- **Final report**: A final report will be delivered to highlight the main accomplishments of the project.

6 Acceptance Criteria

The deliveries will be checked against the items described in this document.

7 Phased Project Plan

Phase 1 (~3 months): survey of the state of the art in FPGA/LLVM

Phase 2 (~3 months): concept formulation

Phase 3 (~6 months): development of the key prototype components

Phase 4 (~6 months): prototype completion

Phase 5 (~3 months): feature completion and tuning

Phase 6 (~3 months): project review and acceptance

Click here to back to the Top Page

HIRPO2017050504: Integrating FPGA Accelerators in Spark by Compiler technology for Heterogeneity

1 Theme: Computing Technology

2 Subject: Heterogeneous computing

List of Abbreviations

FPGA	Field Programmable Gate Arrays

3 Background

There is increasing interest in using Field Programmable Gate Arrays (FPGAs) to accelerate computation, particularly for big data processing on cloud platforms and in data centers. This interest is driven by the growing need for high-performance energy-efficient computing. Today, several vendors have systems that integrate FPGAs into data-center platforms, including ones from IBM, Intel, Xilinx and Microsoft. It is safe to predict that this interest will continue to grow over the next few years and play a central role in the design and use of cloud and data center computing platforms.

One class of FPGA-accelerated systems has recently emerged as a dominant platform for data center acceleration: systems that tightly integrate the FPGA with the processor to share system memory. These systems enable an accelerator circuit implemented on the FPGA fabric to directly read from and write to memory in manner that is coherent with the processor’s caches. A key advantage of this form of integration of is that both processor threads and an FPGA circuit can concurrently perform a computation, sharing data at a fine granularity. This, in turn, can result in higher performance gains and may enable new classes of applications to benefit from FPGA acceleration.

However, there are challenges to exploiting FPGAs for the acceleration of big data applications on these platforms. First, it is unclear the extent to which big data applications can benefit from concurrent processor-FPGA acceleration, given the diverse, and in some cases the irregular, nature of these applications. Second, it is also unclear how to best present FPGA accelerators to programmers. Big data applications are expressed using frameworks such as Hadoop or Spark. Thus, an important concern is how to integrate FPGA accelerators into these frameworks in a way that preserves their APIs, which programmers are already familiar with. Finally, FPGAs are programmed with hardware design abstractions and the associated tools can take excessively long times. This makes the use of FPGAs foreign and tedious to most software and application developers.

The overall goal of this project research is to tackle the above challenges. More specifically, the project aims to explore the benefits and potential overheads of the simultaneous use of processor threads and FPGA circuits to accelerate big data applications, and determine the extent to which such tight integration of the processor and the FPGA can open up opportunities for acceleration of big data applications. Further, this project tries to explore how to best integrate the FPGA accelerator into common data center frameworks.

4 Scope

This proposal primarily focus on following task 4.2 : Integrating FPGA Accelerators in Spark by Compiler technology for Heterogeneity(HIRPO2017050504).

This project consists of the following three specific aspects,

4.1 Acceleration of Graph Analytics based on processor-FPGA system(HIRPO2017050503)

The project will the benefits of tight processor-FPGA integration in the acceleration of big-data graph analytics. In particularly, by focusing on a tightly-coupled processor-FPGA system, (1) it will explore the concurrent use of processor threads and the FPGA accelerator; (2) it will use the shared memory space to alleviate limitations arising from data copying; and (3) it will focus on more application-oriented graph analytics problems.

A FPGA Accelerated Function Units (AFUs) will be designed and implemented to speed up the applications. CPU threads and these AFUs will be used simultaneously, through fine-grain sharing of graph data, to assess (1) the ability of an AFU to accelerate applications with such irregular data access patterns and (2) the benefits of using both processor threads and the AFU to improve performance.

4.2 Integrating FPGA Accelerators in Spark by Compiler technology for Heterogeneity(HIRPO2017050504)

4.2.1 Integrating FPGA Accelerators in Spark

The project will exploit the presence of shared memory between the processor and the FPGA accelerator to achieve better and more transparent integration. The goal is to present the accelerator to the software so it appears as “faster” thread of execution to the JVM runtime, able to fully access data in shared memory. It will be achieved through three mechanisms. In this way, the Spark runtime can appropriately manage the FPGA accelerator, allowing better utilization and load balancing.

4.2.2 Compiling for Heterogeneity

Compiler technology will be developed to ease the use of an FPGA-based heterogeneous system, also in the context of big data processing. Specifically, it will be conducted in two main directions. The first is developing compiler analyses that aid application programmers in determining which segments of

their code are most amenable to FPGA acceleration. These analyses consider not only code characteristics, but also the capabilities of the underlying overlays. The second direction also utilizes overlays to allow the generation of an accelerator circuit for code segments on-the-fly, as threads are scheduled for execution on a target machine. This enables the same code base to execute on either a CPU-only machine or on an FPGA-enabled machine with no explicit modifications/actions by the programmer.

5 Expected Outcome and Deliverables

- **Publications:** 3~5 research publication submissions are expected to document the novelty and results of our research, ~2 publications on the graph analytics component, ~2 publications on integrating FPGAs into the Spark framework, and ~1 publication on the compiling for heterogeneity.
- **Software components:** new LLVM-based compiler passes for application analysis, which will be released to Huawei upon their completion in the course of the project.
- **Accelerated function units**: the designs of these hardware components (in the form of Verilog files and associated documentations) will be also released to Huawei upon their completion in the course of the project.
- **Presentations:** periodical presentations to Huawei to ensure the expected progress and results.
- **Final report**: A final report will be delivered to highlight the main accomplishments of the project.

6 Acceptance Criteria

The deliveries will be checked against the items described in this document.

7 Phased Project Plan

Phase 1 (~3 months): survey of the state of the art in FPGA/LLVM

Phase 2 (~3 months): concept formulation

Phase 3 (~6 months): development of the key prototype components

Phase 4 (~6 months): prototype completion

Phase 5 (~3 months): feature completion and tuning

Phase 6 (~3 months): project review and acceptance

Click here to back to the Top Page

HIRPO2017050505: Memory Management with Mix-Size Pages in Virtualization

1 Theme: Computing Technology

2 Subject: Memory Virtualization

List of Abbreviations

KVM	Kernel-based Virtual Machine
VM	Virtual Machine
THP	Transparent Huge Pages

3 Background

Nowadays, applications use mix-size pages for better performance. Hugepages can deliver better performance than regular 4KB pages in several ways. Linux supports transparent hugepage, which can automatically map hugepages.

In virtualization environments, memory ballooning is a memory management technique used by a hypervisor to allow the physical host system to retrieve unused memory from certain guest VMs and share it with others.

There are many problems of memory management with mixed size pages in some open source virtualization environments, such as KVM and QEMU, which cannot balloon memory with hugepages, and cause hugepage fallback and performance reduction

4 Scope

We are seeking proposals to deal with memory management problems with mix-size pages. To solve hugepage fallback and performance reduction problems, the proposal should implement mix-size pages support for memory management in an open source virtualization environment.

5 Expected Outcome and Deliverables

We expect the outcome and deliverables as following:

- An investigation Report for the differences among the balloon drivers of XEN/KVM/HyperV/VMWare;
- Implementation of mix-size pages support for memory management in an open source virtualization environment;
- Design documents, validation and test reports for the implementation.

6 Acceptance Criteria

As for memory management in virtualization, memory ballooning is a popular technique which allows guests to reduce their memory size (thus providing memory for the host) and to increase it back (thus taking memory from the host). Current balloon drivers in both guests and hosts of most open source virtualization environments can only operate memory pages in 4K normal size. Proposals should meet following requirements:

- The balloon driver can transfer mix-size pages between guests and host.
- The balloon driver can allocate and deallocate memory with mix-size pages.

- The memory management solution is comparable with VMware solution in performance.

7 Phased Project Plan

Phase No.	Phase description	Time(months)	Main task content	Output Standard that should achieve
1	Design and review of proposal.	2	Proposal is accepted by community and approved by Huawei.	Investigation Report Design documents of proposal.
2	Mix-size pages support for memory management in an open source virtualization environment.	6	Implementation of mix-size pages support for memory management in an open source virtualization environment.	The implementation, the validation and test reports.

Click here to back to the Top Page

HIRPO2017050506: Programming models and static/dynamic tools for complex accelerator

1 heme: Computing Technology

2 Subject: Heterogeneity programming model and compilation technology

List of Abbreviations

GPU	Graphics Processing Unit

3 Background

Some recent works (e.g., Catapult project from Microsoft) already show how efficient modern accelerators (e.g., FPGAs) can be in speeding up web search, trading operations and domain-specific compression. But in order to realize the full potential of this hardware, several major challenges needs to be addressed. First, a powerful computational substrate needs to be identified that is suitable for a particular important application. This means that for every domain-specific problem, we have to identify the key characteristics of its computational routines and what type of hardware (e.g., CPUs, GPUs and FPGAs) will best achieve both high performance and high energy efficiency. For example, some of Dr. Pekhimenko's previous works in the field of Bioformatics already show that many genome mapping algorithms can be efficiently mapped to modern GPUs, while some algorithms that require higher flexibility would benefit more from mapping to FPGAs..

Second, there is a need for a flexible programming model and corresponding compiler support to convey characteristics about these applications (e.g., the level of parallelism or type of dependencies) to the actual hardware. Currently, the burden of a proper mapping falls almost entirely on the application developer, which is an important task, requiring both deep understanding of the hardware resources available and the applications algorithm. An even greater challenge arises because the mapping that was good for one generation of the accelerator might not be good for the next one, and hence retuning might be needed.

In the short term, Pekhimenko's group plans to investigate new ways to move most of the mapping and tuning complexity from the programmer (i) to the programming models (both domain-specific and generic) and (ii) to the static/dynamic tools, such as compilers and runtimes. It is extremely important to perform these mapping and tuning mostly automatically, to make sure that regular programmers do not have to deal with complex accelerator programming.

In the longer term, the group will search for new ways (both in hardware and in software) to support the efficient communication, resource sharing, and scheduling between many, potentially very different, architectures within a heterogeneous system. This will be done by carefully exploring the key performance bottlenecks for every application in study, and applying the proper hardware acceleration techniques for the core parts of this application.

4 Scope

This proposal primarily focus on following task 4.1: Programming models and static/dynamic tools for complex accelerator(HIRPO2017050506).

4.1 Programming models and static/dynamic tools for complex accelerator(HIRPO2017050506)

investigate new ways to move most of the mapping and tuning complexity from the programmer (i) to the programming models (both domain-specific and generic) and (ii) to the static/dynamic tools, such as compilers and runtimes. It is extremely important to perform these mapping and tuning mostly automatically, to make sure that regular programmers do not have to deal with complex accelerator programming.

4.2 Software and hardware optimization for heterogeneous system(HIRPO2017050507)

In the longer term, search for new ways (both in hardware and in software) to support the efficient communication, resource sharing, and scheduling between many, potentially very different, architectures within a heterogeneous system.

5 Expected Outcome and Deliverables

- **Publications:** 3~5 research publication submissions are expected to document the novelty and results of our research
- **Software components:** new LLVM-based compiler and some static and dynamic tools
- **Final report**: A final report will be delivered to highlight the main accomplishments of the project.

6 Acceptance Criteria

The deliveries will be checked against the items described in this document.

7 Phased Project Plan

Milestone 1.

Studying the effectiveness of several existing DNN training algorithms for convolutional neural networks (CNN) (as the most popular ones) on modern CPUs and GPUs.

Identifying performance-critical bottlenecks, and potential for compiler/hardware optimization. Studying both single-machine and distributed systems performance.

Milestone 2.

Extending the analysis across different types of DNNs (e.g., RNNs (recurrent neural networks), LSTMs (long short-term memory), SNN (spiking neural networks)).

Identifying performance-critical bottlenecks, and potential for compiler/hardware optimization. Studying both single-machine and distributed systems performance.

Milestone 3.

Memory-level optimizations. Investigating the potential of data compression to reduce memory footprint during training (especially critical for GPUs).

Studying different existing frameworks (e.g., TensorFlow, CNTK, Caffe etc.).

Milestone 4.

Based on Milestone 1-2 analysis, look for potential acceleration using FPGAs. Ideally, build a compiler pass that can automatically select the parts of computation that are amenable for FPGA-based acceleration.

Click here to back to the Top Page

HIRPO2017050507: Software and hardware optimization for heterogeneous system

1 Theme: Computing Technology

2 Subject: Heterogeneity programming model and compilation technology

List of Abbreviations

GPU	Graphics Processing Unit

3 Background

Some recent works (e.g., Catapult project from Microsoft) already show how efficient modern accelerators (e.g., FPGAs) can be in speeding up web search, trading operations and domain-specific compression. But in order to realize the full potential of this hardware, several major challenges needs to be addressed. First, a powerful computational substrate needs to be identified that is suitable for a particular important application. This means that for every domain-specific problem, we have to identify the key characteristics of its computational routines and what type of hardware (e.g., CPUs, GPUs and FPGAs) will best achieve both high performance and high energy efficiency. For example, some of Dr. Pekhimenko's previous works in the field of Bioformatics already show that many genome mapping algorithms can be efficiently mapped to modern GPUs, while some algorithms that require higher flexibility would benefit more from mapping to FPGAs..

Second, there is a need for a flexible programming model and corresponding compiler support to convey characteristics about these applications (e.g., the level of parallelism or type of dependencies) to the actual hardware. Currently, the burden of a proper mapping falls almost entirely on the application developer, which is an important task, requiring both deep understanding of the hardware resources available and the applications algorithm. An even greater challenge arises because the mapping that was good for one generation of the accelerator might not be good for the next one, and hence retuning might be needed.

In the short term, Pekhimenko's group plans to investigate new ways to move most of the mapping and tuning complexity from the programmer (i) to the programming models (both domain-specific and generic) and (ii) to the static/dynamic tools, such as compilers and runtimes. It is extremely important to perform these mapping and tuning mostly automatically, to make sure that regular programmers do not have to deal with complex accelerator programming.

In the longer term, the group will search for new ways (both in hardware and in software) to support the efficient communication, resource sharing, and scheduling between many, potentially very different, architectures within a heterogeneous system. This will be done by carefully exploring the key performance bottlenecks for every application in study, and applying the proper hardware acceleration techniques for the core parts of this application.

4 Scope

This proposal primarily focus on following task 4.2: Software and hardware optimization for heterogeneous system(HIRPO2017050507).

4.1 Programming models and static/dynamic tools for complex accelerator(HIRPO2017050506)

investigate new ways to move most of the mapping and tuning complexity from the programmer (i) to the programming models (both domain-specific and generic) and (ii) to the static/dynamic tools, such as compilers and runtimes. It is extremely important to perform these mapping and tuning mostly automatically, to make sure that regular programmers do not have to deal with complex accelerator programming.

4.2 Software and hardware optimization for heterogeneous system (HIRPO2017050507)

In the longer term, search for new ways (both in hardware and in software) to support the efficient communication, resource sharing, and scheduling between many, potentially very different, architectures within a heterogeneous system.

5 Expected Outcome and Deliverables

- **Publications:** 3~5 research publication submissions are expected to document the novelty and results of our research
- **Software components:** new LLVM-based compiler and some static and dynamic tools
- **Final report**: A final report will be delivered to highlight the main accomplishments of the project.

6 Acceptance Criteria

The deliveries will be checked against the items described in this document.

7 Phased Project Plan

Milestone 1.

Studying the effectiveness of several existing DNN training algorithms for convolutional neural networks (CNN) (as the most popular ones) on modern CPUs and GPUs.

Identifying performance-critical bottlenecks, and potential for compiler/hardware optimization. Studying both single-machine and distributed systems performance.

Milestone 2.

Extending the analysis across different types of DNNs (e.g., RNNs (recurrent neural networks), LSTMs (long short-term memory), SNN (spiking neural networks)).

Identifying performance-critical bottlenecks, and potential for compiler/hardware optimization. Studying both single-machine and distributed systems performance.

Milestone 3.

Memory-level optimizations. Investigating the potential of data compression to reduce memory footprint during training (especially critical for GPUs).

Studying different existing frameworks (e.g., TensorFlow, CNTK, Caffe etc.).

Milestone 4.

Based on Milestone 1-2 analysis, look for potential acceleration using FPGAs. Ideally, build a compiler pass that can automatically select the parts of computation that are amenable for FPGA-based acceleration.

Click here to back to the Top Page