P1-26: Reconfigurable Systems

Mission-Critical Computing
NSF CENTER FOR SPACE, HIGH-PERFORMANCE, AND RESILIENT COMPUTING (SHREC)

SHREC Annual Workshop (SAW25-26)

University of Pittsburgh · BYU Brigham Young University · Virginia Tech · UF University of Florida

January 13-14, 2026

Dr. Alan George
Mickle Chair Professor of ECE
University of Pittsburgh

Dr. Samuel Dickerson
Associate Professor of ECE
University of Pittsburgh

James Bickerstaff
Owen Lucas
Peter Drum
Pavel Serhiayenka
Research Students
University of Pittsburgh

Number of requested memberships ≥ 4

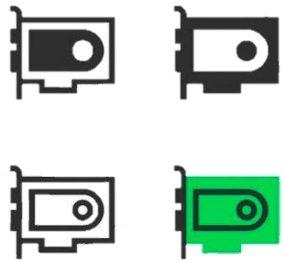
Goals, Motivations, Challenges

Goals

- Develop and evaluate **scalable architectures** for FPGA-based apps
- Compare performance of **spatial and fixed-logic architectures**
- Evaluate FPGA **design tools** for high-level synthesis

Motivations

- FPGAs realize custom datapaths for **efficient processing**
- New processing paradigms required for **increased performance**
- Potential **productivity boost** from high-level design tools



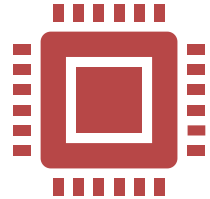
Challenges

- High-level design tools often limit **optimization granularity**
- **SWaP-C constraints** can limit parallelism due to lack of resources
- Potential **resource overhead** produced from high-level design tools

Proposed 2026 Tasks

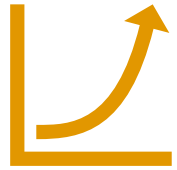
T1: Accelerated Graph Processing

- Expand edge-centric architecture across **multiple operations & memories**
- Investigate **graph neural network inference** using Altera Agilex FPGAs & HBM



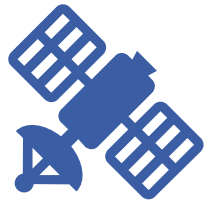
T2: Reliable and Performant Versal Accelerators

- Create ML accelerators and integrate resiliency for **harsh environments**
- Better understand **reliability characteristics** of unique AMD Versal hardware



T3: OPIR Object Tracking

- Identify key kernels in **OPIR object detection** and tracking ground station pipeline
- Accelerate kernels with AMD FPGA to **decrease latency** and **increase throughput**



T4: Distributed-Memory Parallel Graph Analytics

- Explore graph analytic algorithm performance in **parallel-computing architectures**
- Compare **CPU- and NVIDIA GPU-based** design implementations

HBM: High-Bandwidth Memory
ML: Machine Learning
OPIR: Overhead Persistent InfraRed

T1: Accelerated Graph Processing

James Bickerstaff

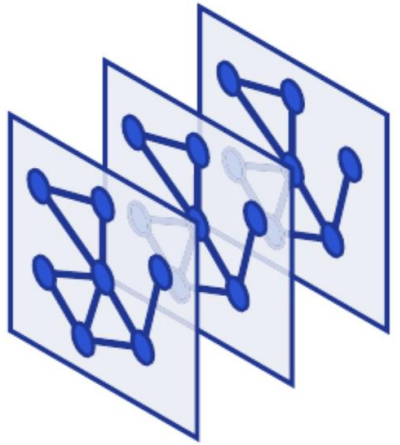
Task Overview

Goals

- Complete optimization of **graph processing architecture** & benchmark across devices
- Create accelerator for **GNN inference** using data-center FPGAs and HLS
- Leverage **HBM and latest Altera & AMD FPGAs** for high-throughput processing

Background

- Graphs provide **flexible** way to represent wide range of datasets & relationships
- **GNNs** growing in popularity & require acceleration to **rapidly process networks**
- **High bandwidth** and **large fabrics** provide opportunity for large-scale parallelism



Approach

- Apply knowledge of **graph partitioning** and **edge-streaming** to GNN application
- Design scalable architecture for maximizing throughput across **HBM & DDR devices**
- Carefully **parameterize system** & **distribute workloads** for scaling beyond single FPGAs

HLS: High-Level Synthesis
HBM: High-Bandwidth Memory
GNN: Graph Neural Network

T2: Reliable and Performant Versal Accelerators

Peter Drum

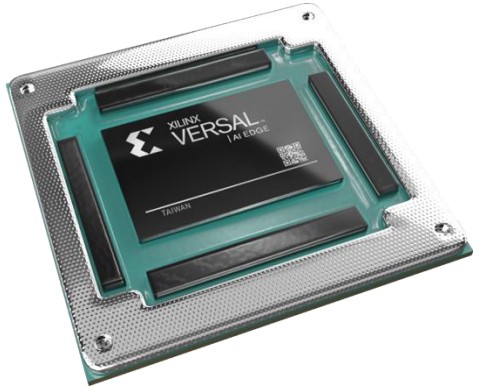
Task Overview

Goals

- Develop and evaluate **machine-learning accelerators** targeted for Versal devices
- Create tools to further understand **Versal hard-IP performance** and **resiliency**
- Investigate **model-specific architectures** for maximized throughput

Background

- Open-source accelerators allow for **more resiliency** to be added to designs
- Testing tools for hard-IP components can help users to better **understand failure modes** of these modalities



Approach

- Continue development of FINN for Versal to improve resiliency and take advantage of **Versal-specific hardware** blocks such as AI Engines and Network-on-Chip
- Develop fault-injection tools for custom Versal **AI Engine fault campaigns**

NoC: Network on Chip

T3: OPIR Object Tracking

Owen Lucas

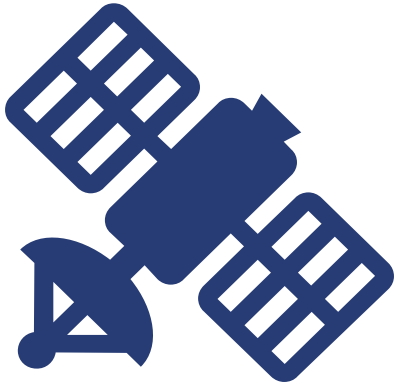
Task Overview

Goals

- Identify key kernels which **bottleneck** terrestrial computing of OPIR data
- Enhance OPIR processing with **heterogeneous architecture** to boost performance
- Leverage **data-center AMD FPGA** cards for large-scale data processing

Background

- Tracking missiles requires rapidly incorporating data from **many sensors**
- **Kalman filters** and **Munkres algorithm** enable multi-hypothesis object tracking
- High-speed nature of missile tracking necessitates **real-time** computation



Approach

- **Vitis High-Level Synthesis** tools allow for rapid FPGA architecture iterations
- Many common computer-vision operations pre-implemented via **Vitis Vision Library**
- OPIR data can be simulated using AFIT Sensor and Scene Emulation Tool (**ASSET**)

OPIR: Overhead Persistent InfraRed
AFIT: Air Force Institute of Technology

Distributed-Memory Parallel Graph Analytics

Pavel Serhiayenka

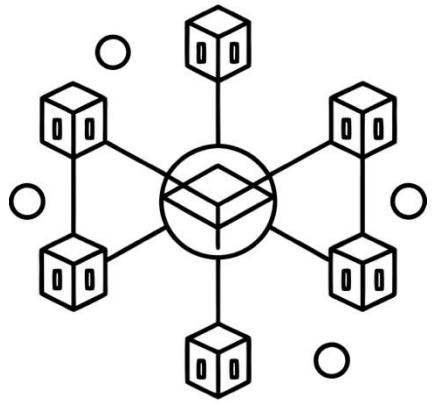
Task Overview

Goals

- Create **distributed-memory implementations** of graph analytic algorithms
- Perform scaling studies of **accuracy versus performance** across many nodes
- Compare **CPU- and GPU-based** implementations with NVIDIA hardware

Background

- **Graph-processing** apps are common workloads for high-performance computing
- Computations can be parallelized between multiple nodes to achieve **speedup**
- **Leiden clustering** provides superior accuracy and performance over **Louvain**



Approach

- Leverage **MPI** to split algorithms over multiple compute nodes
- Utilize modern **GPU programming paradigms** to accelerate graph processing
- Employ **graph modularity** metric to compute and evaluate Leiden algorithm

Milestones, Deliverables, Budget

- Milestones
  - SMW (Summer 2026): Showcase midterm results on all projects
  - SAW (Jan 2026): Demonstrate completion of all projects

- Deliverables
  - Monthly progress reports from all projects
  - Midyear and end-of-year full reports from all projects
  - 4 conference/journal papers (1 per task)
  - Direct access to completed codes and architectures

- Budget
  - **4+** memberships (**200+** votes) for all tasks

Conclusions & Member Benefits

Conclusions

- Design high-throughput architectures for **accelerating graph processing** apps such as GNNs using Altera **data-center FPGAs** and scalable designs for maximizing parallel processing & HBM performance
- Evaluate and improve **machine-learning frameworks** for Versal devices and create custom **fault-injection tools** for understanding failure modes to develop better mitigation strategies
- Develop heterogeneous processing system to offload key **OPIR object tracking** kernels to reduce latency and increase throughput which enables **real-time** missile tracking using multiple sensors
- Employ distributed-memory architectures for **graph analytics** applications such as clustering and node ranking using both **CPU- and GPU-based** designs for maximizing parallel performance

Member Benefits

- **Direct influence** over research direction and projects with frequent meetings
- **Direct benefit** from accelerator designs and tools exploration
- **Direct insights** from research developments, analyses, and publications

GNN: Graph Neural Network
OPIR: Overhead Persistent InfraRed