GPU Ocelot: Dynamic Compilation for PTX
Andrew Kerr, Gregory Diamos, Nails Farooqui, Jeff Young, and Sudhakar Yalamanchili
Georgia Institute of Technology
{arkerr@gatech.edu, gtt250v@mail.gatech.edu, nai@gatech.edu, jyoung9@gatech.edu, sudha@ece.gatech.edu}

Project Goals
- Efficient execution of data-parallel kernels on heterogeneous platforms
- CUDA on multiple architectures: NVIDIA GPU, Multicore CPUs, Vector ISAs, and AMD GPUs
- Performance scalability and portability
- Developer Productivity

Sponsors: NSF, LogicBlox, NVIDIA

Ocelot Overview
- Ocelot - CUDA Runtime API
- Dynamic Compiler
- Translator
- Execution manager
- PTX Emulation
- CUDA and AMD GPUs
- LLVM Translation
- Execution on several architectures
- Instrument and profile kernel execution
- Explore novel GPU architectures

Oncilla Hardware Infrastructure
- Prototype hardware to support non-coherent Global Address Spaces for accelerated data warehousing applications
- Oncilla will support efficient data movement through low-latency packet operations between nodes using HT and EXTOLL interconnects
- Collaboration with University of Heidelberg, Polytechnic University of Valencia, AIC Inc., LogicBlox Inc.
- Sponsors: NVIDIA

Performance Modeling [2]
- Application metrics collected via: static analysis, instrumentation, emulation
- Correlated metrics detected via Principal Component Analysis
- Clusters of applications and machine models identified
- Statistical performance model: predicted performance within 10%

Non-GPU Execution Targets [3]
- Multicore CPU [3]
- Subkernel Formation
- Execute each CTA on a processor core
- Serialize threads within CTA, switch context at CTA-wide barriers
- Explore novel thread scheduling techniques
- Partition kernels into subkernels
- Translate subkernels lazily
- Schedule subkernels on different processors or functional units

Vectorized Multicore Execution [6]
- Transform scalar kernel into vectorized kernel
- Execution of a control path is logically equivalent to executing several PTX threads
- Detect control divergence and exit to execution manager

Workload Characteristics, Optimization, and Productivity Tools [1,4,5,7]
- PTX Emulation [1,4]
- Thread Frontiers [5]
- Checkpointing [5]
- Kernel Instrumentation [7]

References