

ASPLOS 三日目

本会議の一日目 .

Keynote: Developing our Quantum Future

cf. <https://www.microsoft.com/en-us/quantum/development-kit>

- applications
 - chemistry - efficient fertilizer production, mitigation of global warming
 - materials - lossless power lines, better batteries, smart materials
 - machine learning - faster training, improved models
 - optimization - healthcare diagnostics, traffic reduction
- cont. applications
 - Quantum-safe privacy - QKD, communication, Networking, post-Quantum crypto
 - Quantum sensing - biology, medicine, GPS, accelerometry, etc.
 - Quantum games - learn superposition, entanglement, interference,
 - Quantum speedups - semi-definite programming, linear systems of equations
- Topological Quantum Computation
 - <https://arxiv.org/abs/quant-ph/0101025>
 - topology = properties that are insensitive to deformations (i.e., errors) in local geometry
 - no local measurement can measure if the rope is knotted.
 - information encoded in knots is immune to local measurement
- empowering the quantum revolution
 - a quantum software stack maps a quantum-accelerated programs to a hybrid quantum system
- Microsoft Quantum Development Kit
 - <https://www.microsoft.com/en-us/quantum/development-kit>
- Developing Quantum Applications
 - Find quantum algorithm with quantum speedup <- starting point
 - confirm quantum speedup after implementing all I/O and gate operations
 - optimize code until runtime is short enough
 - embed into specific hardware
- Examples: quantum chemistry of FeMoco
 - Quantum algorithm (2012) - 30,000 years
 - Quantum algorithm (2015) - 1.5 days
- Q#

```
operation nextRandomBits(): Result {
    mutable result = Zero;
    using(qubits = Qubit()) {
        H(qubits[0]);
        set result = M(qubits[0]);
        Reset(qubits[0]);
    }
    return result;
}
```

- cf. <https://docs.microsoft.com/quantum/concepts/the-qubit?view=qsharp-preview>
- cf. <https://github.com/Microsoft/Quantum>
 - Functors
 - Type-parameterized functions and operations
 - partial application
- cf. <https://cloudblogs.microsoft.com/quantum/2018/07/23/learn-at-your-own-pace-with-microsoft-quantum-katas/>

Data Movement I

A Framework for Memory Oversubscription Management in Graphics Processing Units

- Eviction, Throttling and Compression selectively for different applications
- アプリでメモリアクセスパターンが違って Oversubscription 対策も違う
 - 3dconv -- striming access, small working set -> hiding eviction latency
 - lud -- data reuse by kernels, small working set -> hiding eviction latency
 - atax -- random access, large working set -> reducitn working set size

Swizzle Inventor: Data Movement Synthesis for GPU Kernels

Swizzle Inventor - swizzle な GPU プログラムを合成してくれる .

- <https://github.com/mangpo/swizzle-inventor>

Scalable Processing of Contemporary Semi-Structured Data on Commodity Parallel Processors A
Compilation-based Approach

- semi-structure data - flexible data mode, "nested", XML,JSON,etc.
 - JSON-family data
- JSON の stream processing では automata ベースの方法がとれない
 - match query, record states, recognize syntax structure
- streaming compilation
 - query set, JSON grammer
 - DFA + pushdown automaton -> streaming automaton
- parallelizing compilation
 - path explosion に対応できるように syntax を変更する
- JPStream
- <https://github.com/AutomataLab/JPStream>

Data Movement II

Buffets: An Efficient and Composable Storage Idiom for Explicit Decoupled Data Orchestration

- アクセラレータ作る時バッファ個別につくるの大変
- バッファを整理して "FIFO" みたいな Idiom として Buffet を提案
- Verilog の実装 - <https://github.com/cwfletcher/buffets>
- データオーケストレーション方法を Implicit/Explicit , Coupled/Decoupled のマトリックスで分類
- Buffet は , E.D.D.O 想定

HiWayLib: A Software Framework for Enabling High Performance Communications for Heterogeneous Pipeline Computations

- a communication library for heterogeneous pipeline computations
- Cumbersome inter-device data movements
 - lazy reference-based scheme
 - region-based lazy data copy
 - reference based task queue
- End detection of pipeline processing
 - Late triggered inter-stage tracking
- Contentions on communication data structure

- Bi-Layer contention relief

StreamBox-HBM: Stream Analytics on High Bandwidth Hybrid Memory

<https://thexsel.github.io/p/streambox/>

- background - <https://www.domo.com/learn/data-never-sleeps-6>
- StreamBox-HBM, 3D メモリ DRAM 向けのストリームエンジン .
 - 110 million records per second and 238 GB/s memory bandwidth
- challenges
 - hash grouping performs poorly on 3D memory
 - parallel sort for grouping, sort outperforms hash on 3D memory
 - 3D memory is capacity limited
 - only use 3D memory for in-memory index
 - how to dynamically map data/operators
 - balance two limited resources
- Evaluation
 - Yahoo stream benchmark で Flink@KNL が 10MRes/s StreamBox-HBM は 50MRes/s .

Potpourri

CASCADE Just-In-Time Compilation for Verilog A New Technique for Improving the FPGA Programming Experience

- Just-in-time
 - run code in a simulator
 - compile in the background
 - translate when finished

<https://github.com/vmware/cascade>

DCNS: Automated Detection Of Conservative Non-Sleep Defects in the Linux Kernel

- waiting operation
 - non-sleep operations
 - sleep-able operations
- mdelay を msleep に , GFP_ATOMIC を GFP_KERNEL に
- function pointer analysis

A Case for Lease-Based, Utilitarian Resource Management on Mobile Devices

Potpourri(3): Android アプリの無駄な電力消費を削減するランタイムの話
cf. <https://orderlab.io/LeaseOS/>