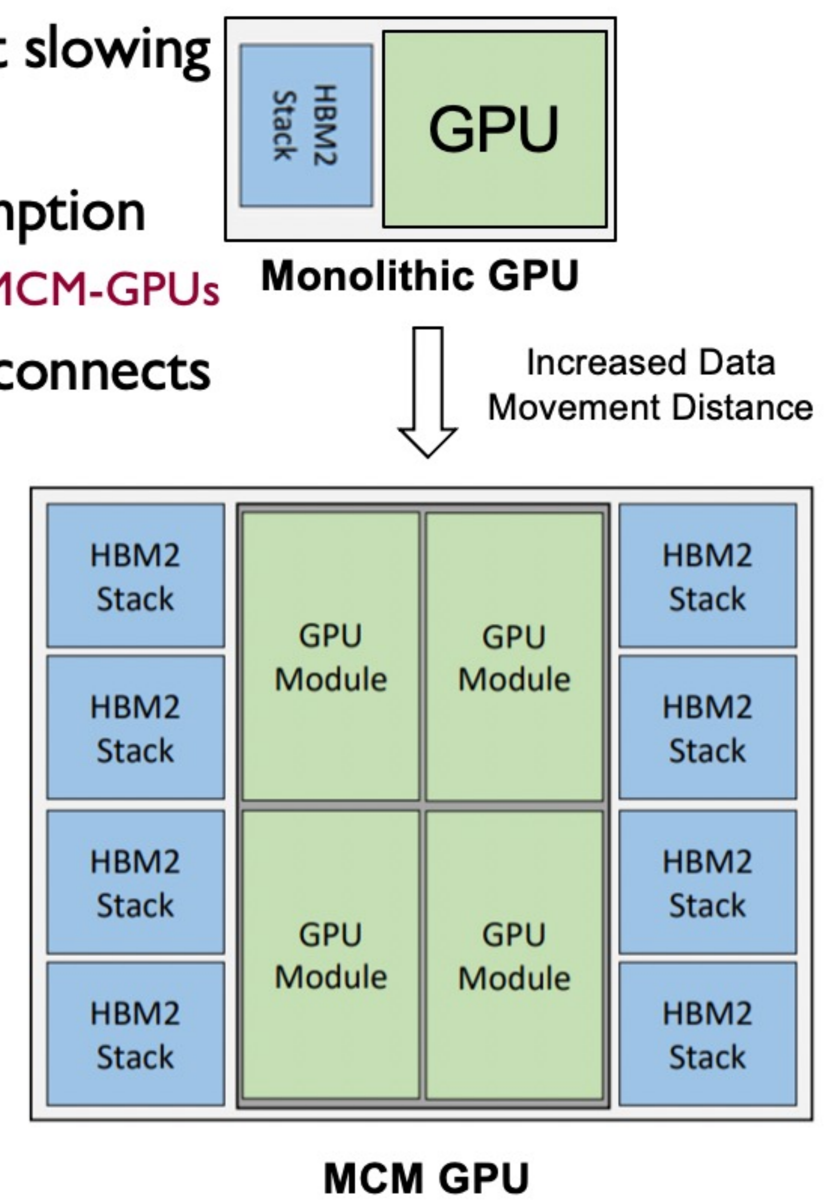
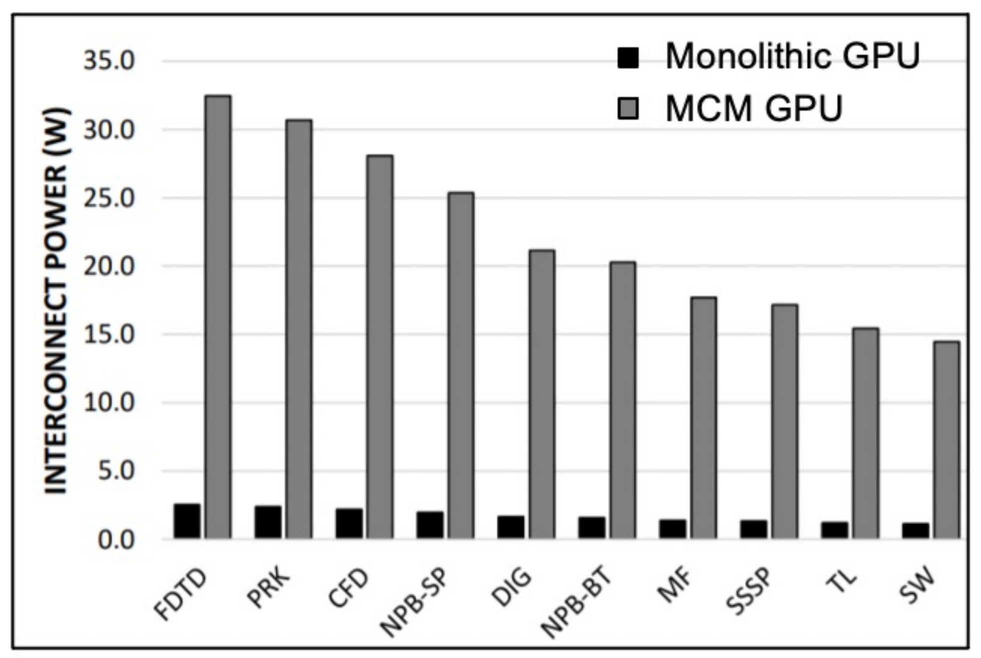
Approximate Pattern Matching for On-chip Interconnect Traffic Prediction

Vignesh Adhinarayanan, Wu-chun Feng | {avignesh, wfeng}@vt.edu

Introduction

- Motivation: Adoption of MCM-GPUs to combat slowing of Moore's law
- Problem: Increased interconnect power consumption
  - Why? Average data movement distance increases on MCM-GPUs
- Solution: (Proactive) DVFS techniques for interconnects





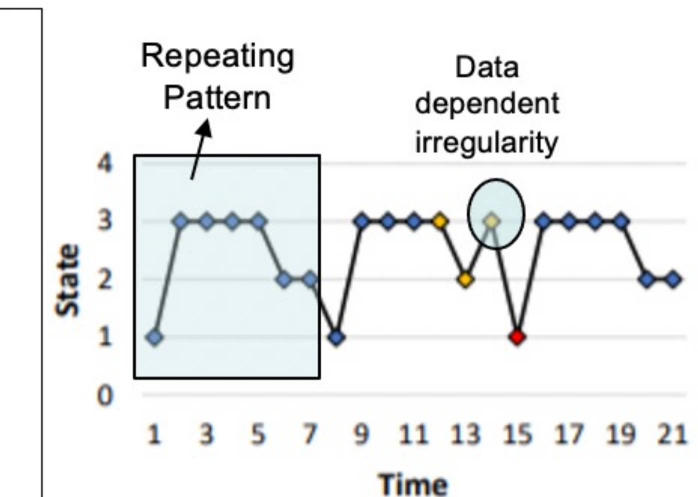
Order of magnitude increase in interconnect power

Proactive Power Management

- **Approach**
  - Predict interconnect traffic for upcoming kernels from past history
  - Set interconnect's P-state based on expected traffic
- **Phase prediction techniques and limitations**
  - **Markov Model:** Slow adaptation to global phase change
  - **History Table:** High mispredictions for applications with irregular traffic and noisy traffic

```
for(i=0; i<1000; i++){
  pre_process(vec);
  val1 = process(MatA);
  val2 = process(MatB);
  val3 = process(MatC);
  val4 = process(MatD);
  d1 = det(MatD);
  if(d1 == 0)
    res = compute(MatA, MatB, MatC, vec);
  else
    res = compute(MatA, MatB, MatD, vec);
}
```

Toy Example



Active Window: 3 | 2 | 3 | 1

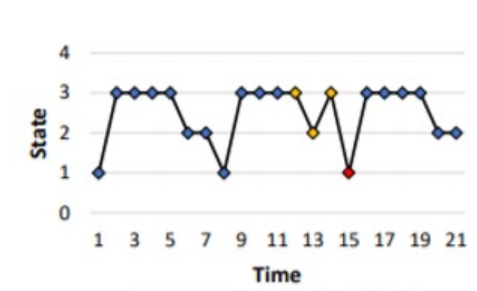
Pattern	Outcome
1333	3
3333	2
3332	2
3322	1
3221	3

No Matches for multiple kernels

VirginiaTech Invent the Future
synergy.cs.vt.edu

Proposed Solution

- Approximate Pattern Matching (APM) History Table
  - Uses edit-distance to find approximate matches
  - Other optimizations: Pattern length tuning, replacement algorithm, edit-distance threshold tuning
- Prediction Error Results
  - 2.66% for approximate matching vs. 3.83% for exact matching (and 7.11% for Markov model)

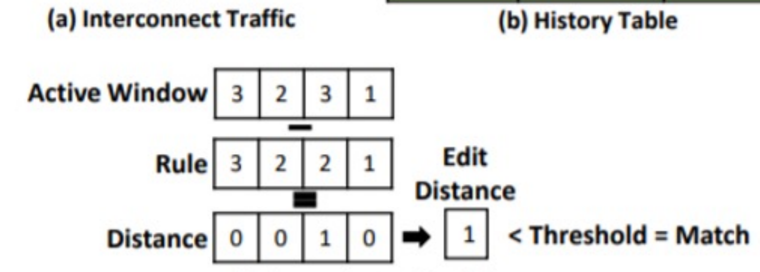


(a) Interconnect Traffic

Active Window: 3 | 2 | 3 | 1

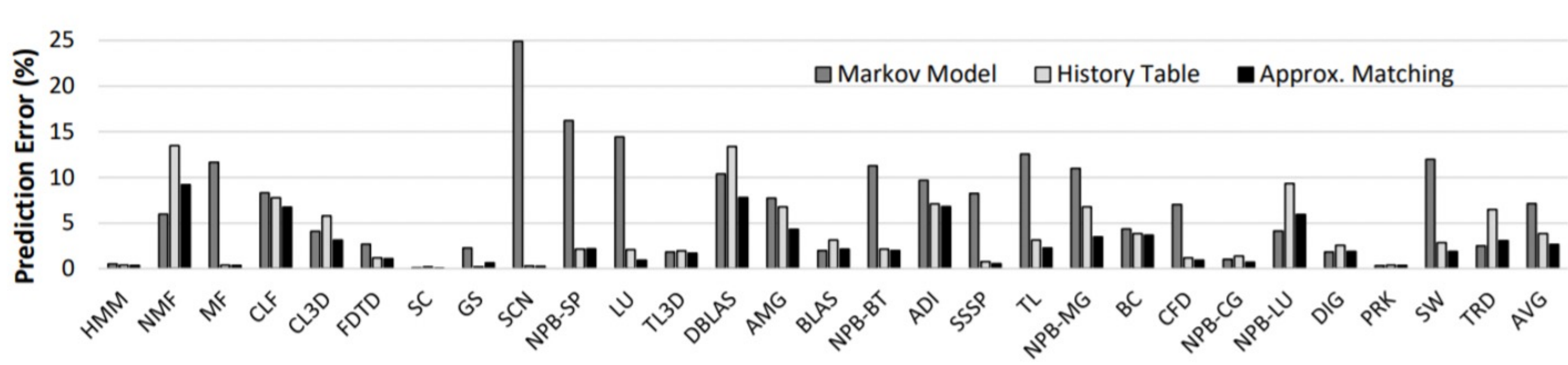
Pattern	Outcome	Frequency
1333	3	2
3333	2	2
3332	2	2
3322	1	1
3221	3	1

(b) History Table



(c) Approximate Matching

Near matches result in successful predictions



VirginiaTech Invent the Future
synergy.cs.vt.edu

Next Steps

- **Conclusion**
  - Phase prediction via approximate pattern matching benefit kernels exhibiting irregular, non-uniform, noisy traffic
    - 3.83% error for state of the art → 2.66% for proposed approach
- **Next Steps**
  - Extensions to approximate pattern matching for different types of mismatches (e.g., inserts, deletes, and swaps)
  - Hardware design and implementation to meet target latency (5us) and power budget (0.1W) for real-world adoption
  - Application-level performance and power evaluations

Authors' Contact Information
Vignesh Adhinarayanan (avignesh@vt.edu)
Wu-chun Feng (wfeng@vt.edu)

The authors thank Joseph Greathouse (AMD) for providing the input traces used in this study.

VirginiaTech Invent the Future
synergy.cs.vt.edu