Approximate Computing for the Internet of Things: from Circuits to Applications

Xun Jiao

Dept. of ECE

Villanova University

Error Tolerant IoT Applications

Approximate Computing exploits application error tolerance to trade accuracy for improved efficiency/performance.

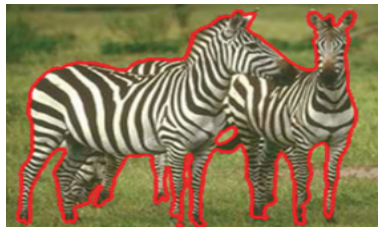
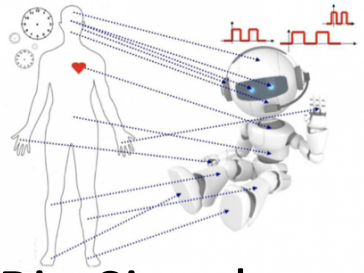


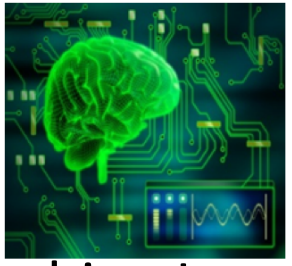
Image Processing



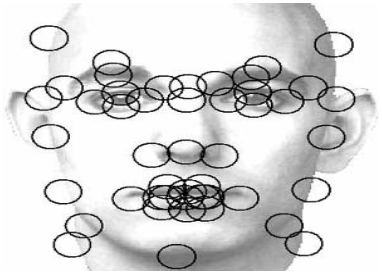
Bio Signals

100% accuracy not always required

Voice Recognition



Machine Learning



Computer Vision

Approximate Computing: Challenges

- Approximate Computing: a cross-layer effort
  - Approximations are performed at different computing levels
  - Output quality is checked at application level

- Challenge: guarantee output quality (under unseen data)
  - Need input-aware error models
    - e.g., for an approximate circuit, are there errors for 1+1? 2+2? a+b?
  - Need controllable/reconfigurable approximations
    - e.g., for an approximation method, how to balance energy-accuracy tradeoff?

Outline

- Timing Error Modeling
- Approximate Computation Reuse
- Future Work

Outline

- Timing Error Modeling
- Approximate Computation Reuse
- Future Work

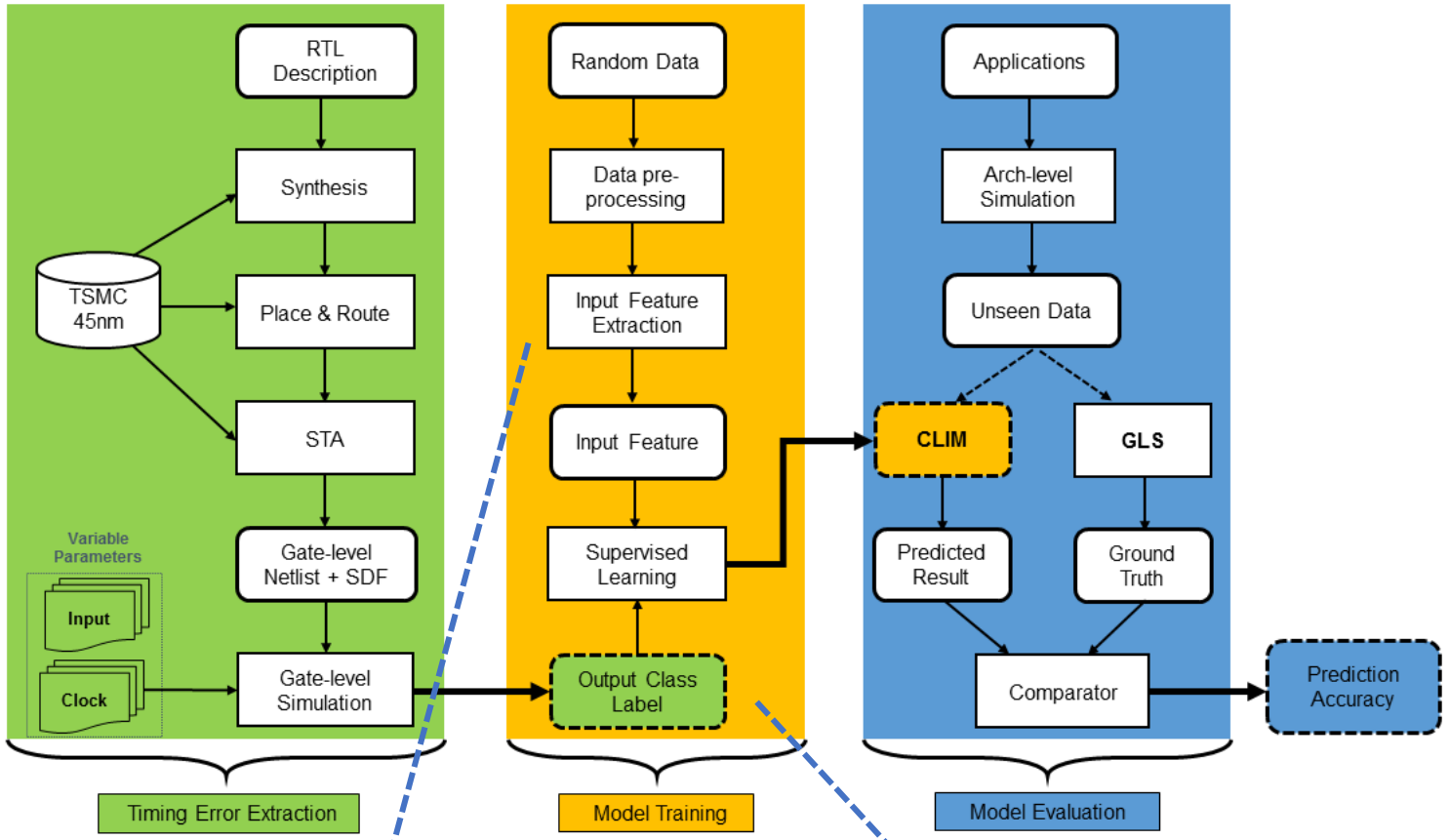
Timing Errors Caused by Frequency Scaling

- Timing errors are an (unwanted) byproduct of approximation methods such as frequency overscaling.

- Model and expose errors to application level for quality assessment
  - Simulation is expensive in *time*, *license, not accessible to SW developers*
  - We propose a machine learning-based model that can predict timing errors for any unseen input:
    - For example, given an adder, predict timing errors for 1+1? 2+2? a+b? running @ 1GHz? 1.5GHz? xGHz?

Xun Jiao, et.al CLIM: A Cross-level Workload-aware Timing Error Prediction Model for Functional Units. *IEEE Transactions on Computers (TC)* 2018.

Machine Learning Modeling



$$I = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & \dots & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & \dots & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & \dots & 1 & 1 & 1 \end{bmatrix} \qquad O_i = \begin{bmatrix} E \\ C \\ E \end{bmatrix}$$

Xun Jiao, et.al CLIM: A Cross-level Workload-aware Timing Error Prediction Model for Functional Units. *IEEE Transactions on Computers (TC)* 2018.

Results

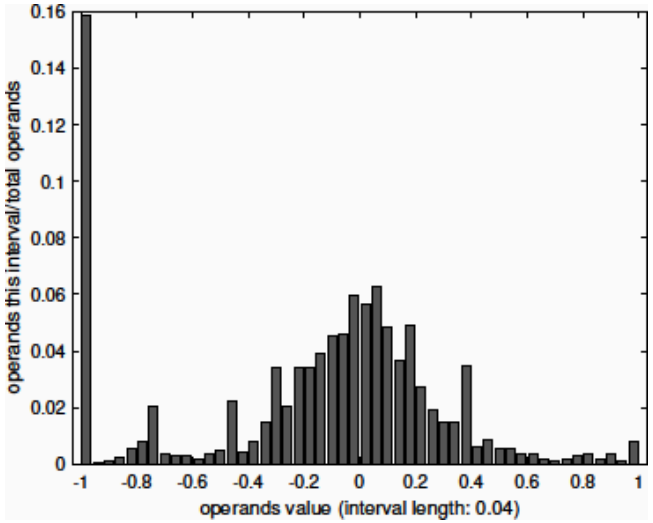
- Error Prediction Accuracy
  - 96% prediction accuracy for timing errors.
- Application quality estimation
  - For example, predict whether an output image quality is acceptable (PSNR > 30dB) under a certain approximation setting
  - 97% estimation accuracy for image applications

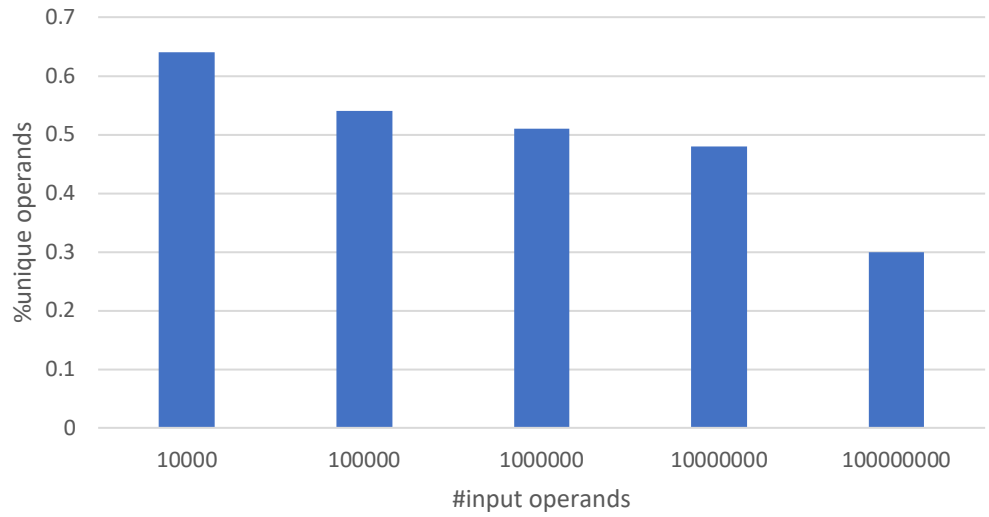
Outline

- Timing Error Modeling
- Approximate Computation Reuse
- Future Work

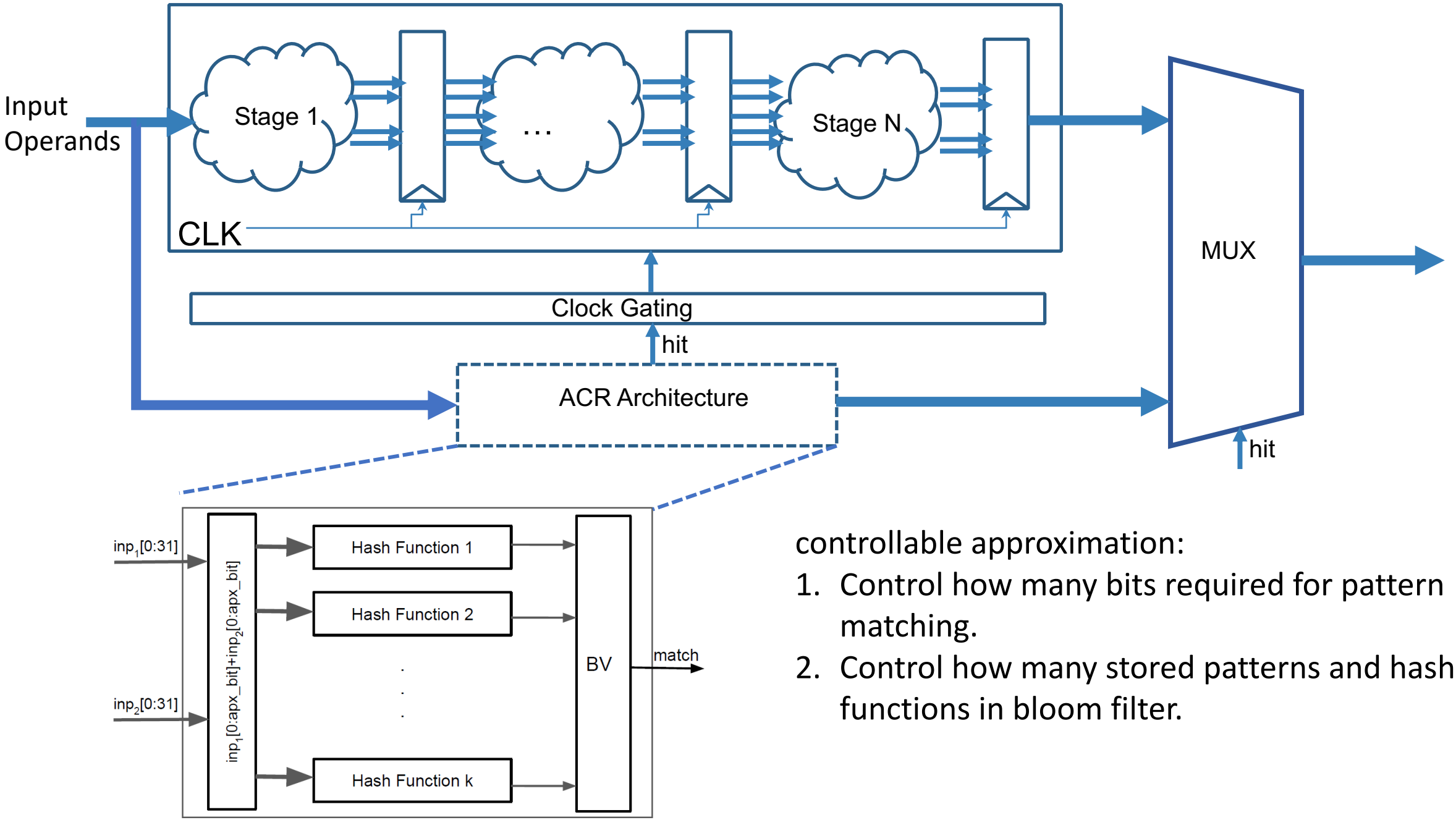
Approximate Computation Reuse

- Observation: Many data-intensive applications show repetitive computations, i.e., computations with same/similar operands.
- Approach: Approximately reuse previous computations
  - pre-store frequent computations, reuse results if matched approximately
    - Use result of (2 * 3) for (1.99 * 3.01)





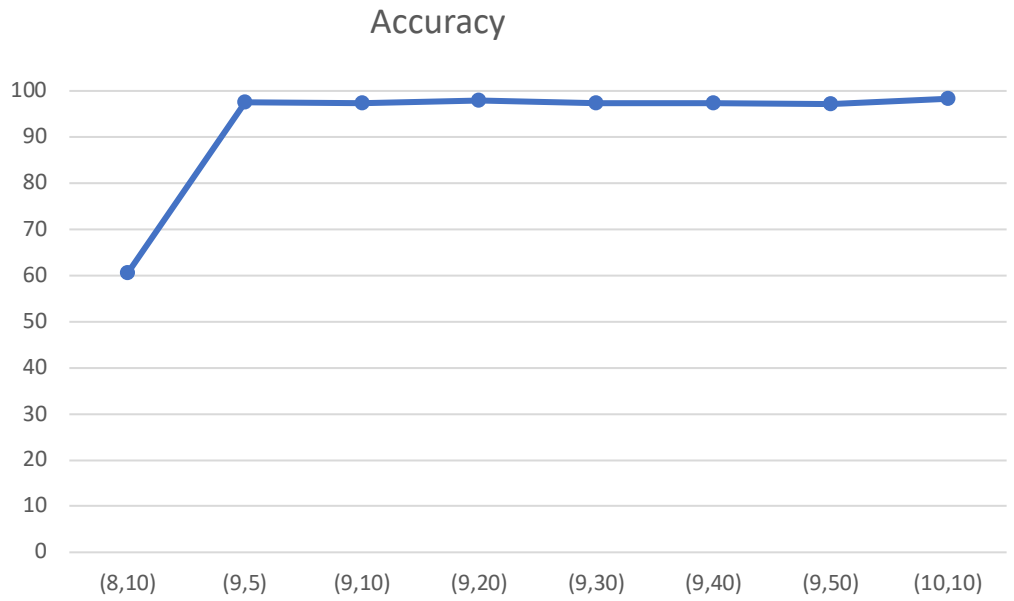
Xun Jiao, et.al Energy-Efficient Neural Networks using Approximate Computation Reuse. IEEE/ACM *DATE* 2018.

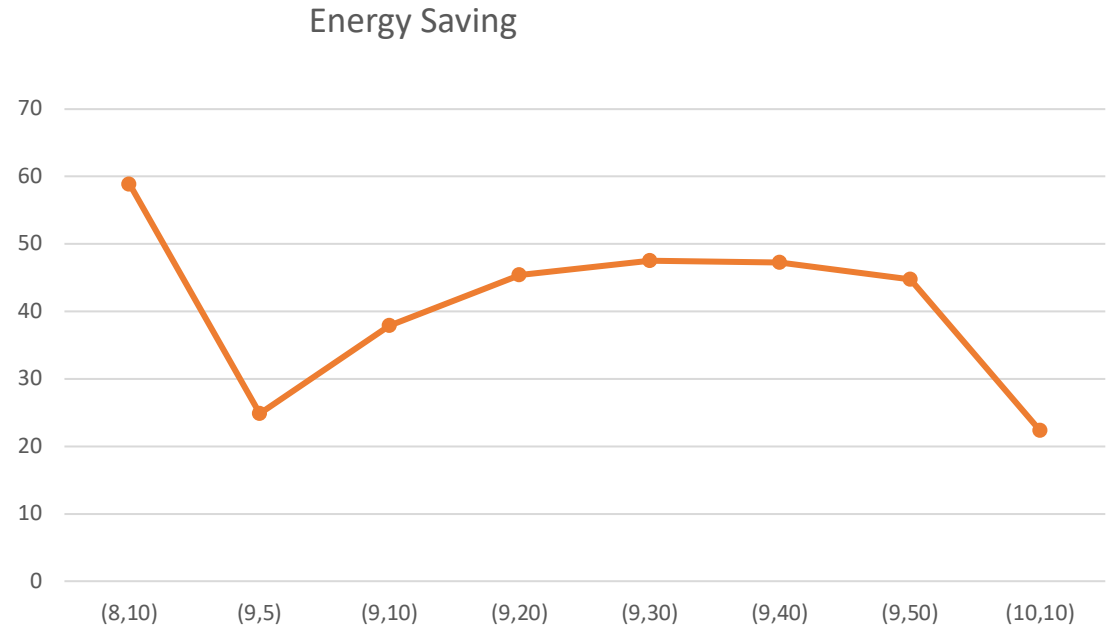


controllable approximation:

1. Control how many bits required for pattern matching.
2. Control how many stored patterns and hash functions in bloom filter.

Accuracy-Energy Tradeoff





Matching Mode = (9,30) Accuracy loss < 1% Avg. Energy Saving: 47.5%

12

Future Work

- Challenge: guarantee output quality (under unseen data)
  - Need an accurate model of approximation-induced errors
  - Need an accurate error injection framework

- Challenge: controlling heterogeneous approximate computing
  - Different methods, different degrees on different units and code blocks
  - Maximize efficiency under quality constraints