Motivation
- Exorbitant amount of data
- The high cost of energy for data movement
- A paradigm shift towards processing close to the data i.e., near-memory computing (NMC)
- However in early design-stage, simulation are extremely slow, imposing long run-time

NAPEL: Performance Prediction via Ensemble Machine Learning
- Fast and accurate performance and energy prediction for a previously-unseen application
- Microarchitecture-independent characterization with architectural simulation responses to train an ensemble algorithm
- Intelligent statistical techniques to extract meaningful data with minimum experimental runs

Phase 1: LLVM Kernel Analysis
- Microarchitecture-independent kernel analysis to generate an application profile independent of the NMC architecture

<table>
<thead>
<tr>
<th>Application Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instruction Mix</td>
<td>The fraction of instruction types (integer, floating point, memory, etc.)</td>
</tr>
<tr>
<td>ILP</td>
<td>Instruction-level parallelism on an ideal machine</td>
</tr>
<tr>
<td>Data/Instruction reuse distance</td>
<td>For a given distance δ, probability of reusing one data element/instruction (in a certain memory location) before accessing δ other unique data elements/instructions (in different memory locations)</td>
</tr>
<tr>
<td>Memory traffic</td>
<td>Percentage of memory reads/writes that need to access the main memory, assuming a cache of size equal to the maximum reuse distance</td>
</tr>
<tr>
<td>Register traffic</td>
<td>An average number of registers per instruction</td>
</tr>
<tr>
<td>Memory footprint</td>
<td>Total memory size used by the application</td>
</tr>
</tbody>
</table>

Phase 2: Central Composite Design
- Design of experiment techniques are used to reduce the number of experiments to train NAPEL
- Central composite design (CCD) is applied to minimize the uncertainty of a nonlinear polynomial model that accounts for parameter interactions
- In CCD, each input parameter can have five levels: min, low, central, high, maximum

Phase 3: Ensemble Machine Learning
- We employ random forest (RF) as our ML algorithm, which embeds procedures to screen input features
- With hyper-parameters tuning to optimize the accuracy of ML algorithm

NAPEL Prediction
- Cross-platform prediction of a completely unseen application by only using micro-architectural independent application features
- 220x faster, on average, than our NMC simulator (min. 33x, max. 1039x)

NMC Architecture
- IBM POWER9
- Ramulator-PIM

Evaluation
- MRE of 8.5% and 11.6% for performance and energy prediction
- NAPEL is 1.7x (1.4x) and 3.2x (3.5x) better in terms of performance (energy) estimation than ANN and decision tree

NMC Suitability Analysis
- NAPEL provides an accurate prediction of NMC suitability
- MRE between 1.3% to 26.3% (average 14.1) for EDP prediction
- Workloads with EDP<1, are not suitable for NMC and can leverage the host cache hierarchy

References