### Introduction

- Optimizing energy consumption is a challenge for HPC systems.
- READEX provides an auto-tuning framework to tune applications for energy-efficiency.
- Exploits dynamic behavior of applications w.r.t. execution time and compute intensity.
- **Intra-phase**: variation in the control flow within a phase
- **Inter-phase**: variation in the execution characteristics between phases
  - Leverages similar phase behavior

### READEX Tools-Aided Methodology

**Design Time Analysis (DTA):**
- Performed by the Periscope Tuning Framework (PTF).
- Detects coarse-granular program regions.
- Tuning plugin explores system configurations.
- Supports:
  - CPU frequency, uncore frequency, OpenMP threads tuning parameters
  - Energy, execution time, CPU energy, EDP, ED2P, and TCO tuning objectives
  - Requests measurements for runtime situations (rts’s) of significant regions and the phase.
  - Performs one/more tuning steps.
  - Best system configurations for the rts’s are stored in a tuning model.

**Runtime Application Tuning (RAT):**
- Loads the tuning model.
- Dynamically switches to the best configuration for a detected rts during production runs.
- Performs calibration for unseen rts’s.

### Inter-Phase Tuning Plugin

- Performs three tuning steps.
- **Cluster analysis:**
  - Creates a search space using the random search strategy for the selected objective.
  - Uses DBSCAN to cluster phases.
    - Normalized compute intensity (#AVX_Instructions/1.5 cache misses) and conditional branch instructions as features.
  - Selects cluster-best configurations for lowest objective value normalized by #AVX_Instructions.
- **Default Execution:**
  - Each experiment measures the objective value for the default system configuration.
- **Verification:**
  - Runs experiments with cluster-best configurations for the phase and the rts’s.
  - Computes static savings for the phase, static savings for rts’s, and dynamic savings for rts’s.

### Evaluation - Savings

**miniMD:**
- Performs molecular dynamics simulation of a Lennard-Jones EAM system.

**INDEED:**
- Sheet metal forming simulation software.
- Performs an adaptive mesh refinement.

<table>
<thead>
<tr>
<th>Application</th>
<th>Static savings for the whole phase (%)</th>
<th>Static savings for the rts’s (%)</th>
<th>Dynamic savings for the rts’s (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>miniMD</td>
<td>13.74</td>
<td>14.51</td>
<td>0.03</td>
</tr>
<tr>
<td>INDEED</td>
<td>5.75</td>
<td>9.24</td>
<td>10.45</td>
</tr>
</tbody>
</table>

### Discussion:

- Static savings of 13.74% for miniMD and 5.75% for INDEED.
- miniMD records lower dynamic savings.
- miniMD has only two significant regions, while INDEED has nine significant regions.
- INDEED has more potential for dynamism, and hence, better dynamic savings.

### Conclusions

- **READEX** performs runtime tuning guided by a tuning model, which is computed at design-time.
- The *READEX_interphase* plugin exploits dynamism by:
  - Using normalized PAPI metrics for clustering phases based on their execution characteristics.
  - Selecting cluster-best configurations for phase and rts’s.
- Savings highlight the effectiveness of this methodology.
- Shows great promise in tuning for energy-efficiency.

---

**Figure 1:** Variation of the compute intensity w.r.t. the iteration number of the time loop in (a) miniMD, and (b) INDEED.