THE UNIVERSITY OF TEXAS AT AUSTIN

Electrical and Computer Engineering

Automatic Calibration of Micro-Architecture Description Models

[†]Zhuoran Zhao, [‡]Gary Morrison and [†]Andreas Gerstlauer

[†]Electrical and Computer Engineering, The University of Texas at Austin

[‡]Freescale Semiconductor Inc.

SRC Semiconductor Research Corporation

global RESEARCH COLLABORATION

Introduction

Motivation

- Increasing design complexities
  - Rapid system design space exploration
  - ➢Fast generation of accurate Instruction Set Simulator (ISS)
- Traditional approaches
  - Open source ISS, e.g. gem5, OVP …
  - Manually built processor models in C++
  - ➢Inaccurate or too hard to explore new features
- ➢Automatic performance model generation
  - ➢Parameterize micro-architecture description model
  - ➢Calibrate parameterized micro-architecture model

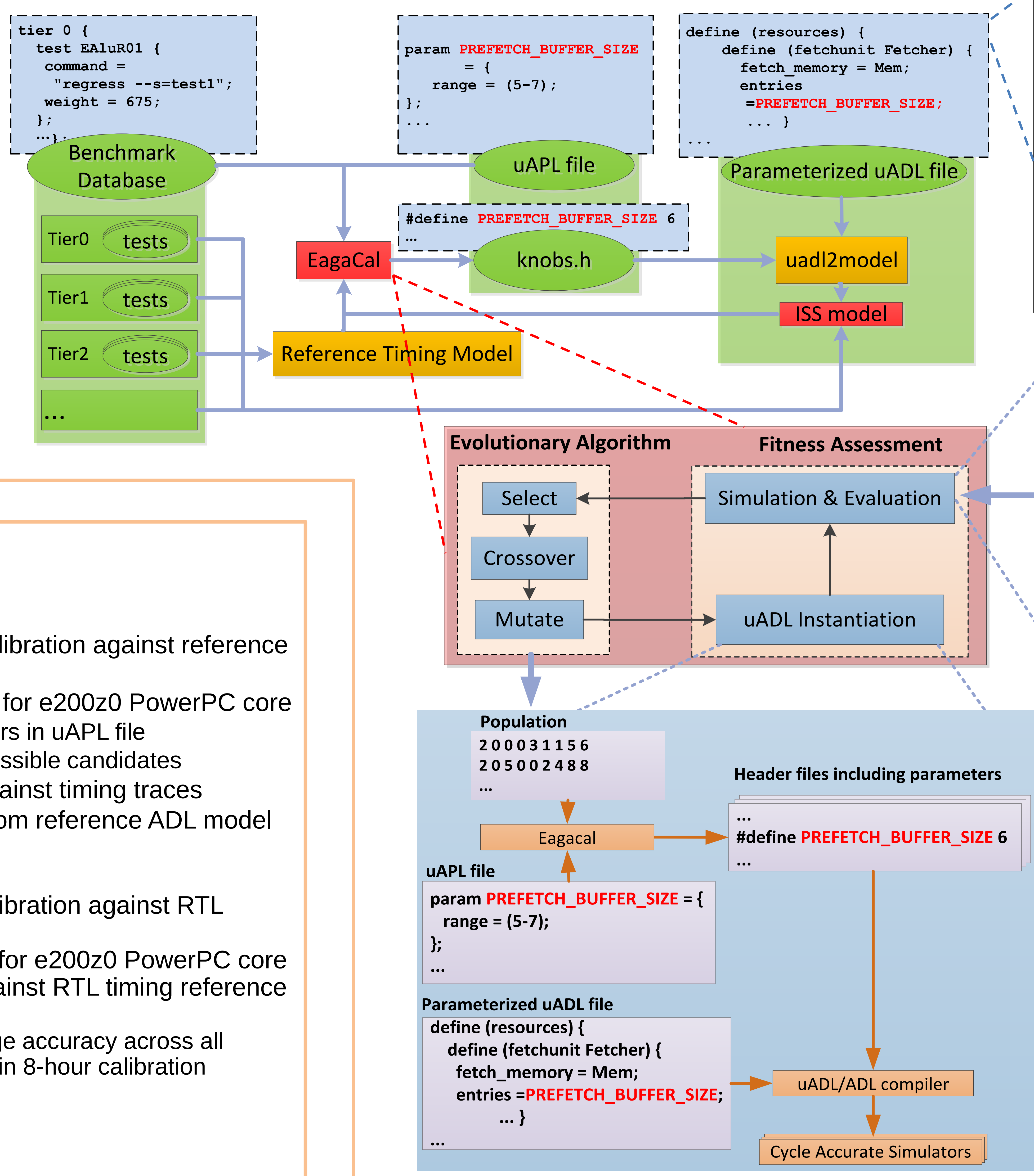
Challenges and Solutions

- How to efficiently model micro-architecture performance?
  - Easy to be parameterized
  - Reflect real structures for further exploration
  - ➢Freescale open source Architecture/Micro-architecture Description Language (ADL/uADL)
- How to speed up the searching strategy?
  - Multi-tiered benchmark sets
  - Hierarchically calibrate various aspects

Automatic Calibration Framework

Evolutionary Calibration Infrastructure

- Evolutionary/Genetic Algorithm CALibration (EagaCal)
  - Calibrate manually parameterized uADL model
  - Parameters specification in uAPL file format
  - Multi-tiered benchmark set



```
tier 0 {
  test EAluR01 {
   command =
    "regress --s=test1";
   weight = 675;
  };
  …};
```

```
param PREFETCH_BUFFER_SIZE
       = {
   range = (5-7);
};
...
```

```
define (resources) {
    define (fetchunit Fetcher) {
      fetch_memory = Mem;
      entries
       =PREFETCH_BUFFER_SIZE;
       ... }
...
```

```
#define PREFETCH_BUFFER_SIZE 6
…
```

Population

```
2 0 0 0 3 1 1 5 6
2 0 5 0 0 2 4 8 8
…
```

Header files including parameters

```
…
#define PREFETCH_BUFFER_SIZE 6
…
```

uAPL file

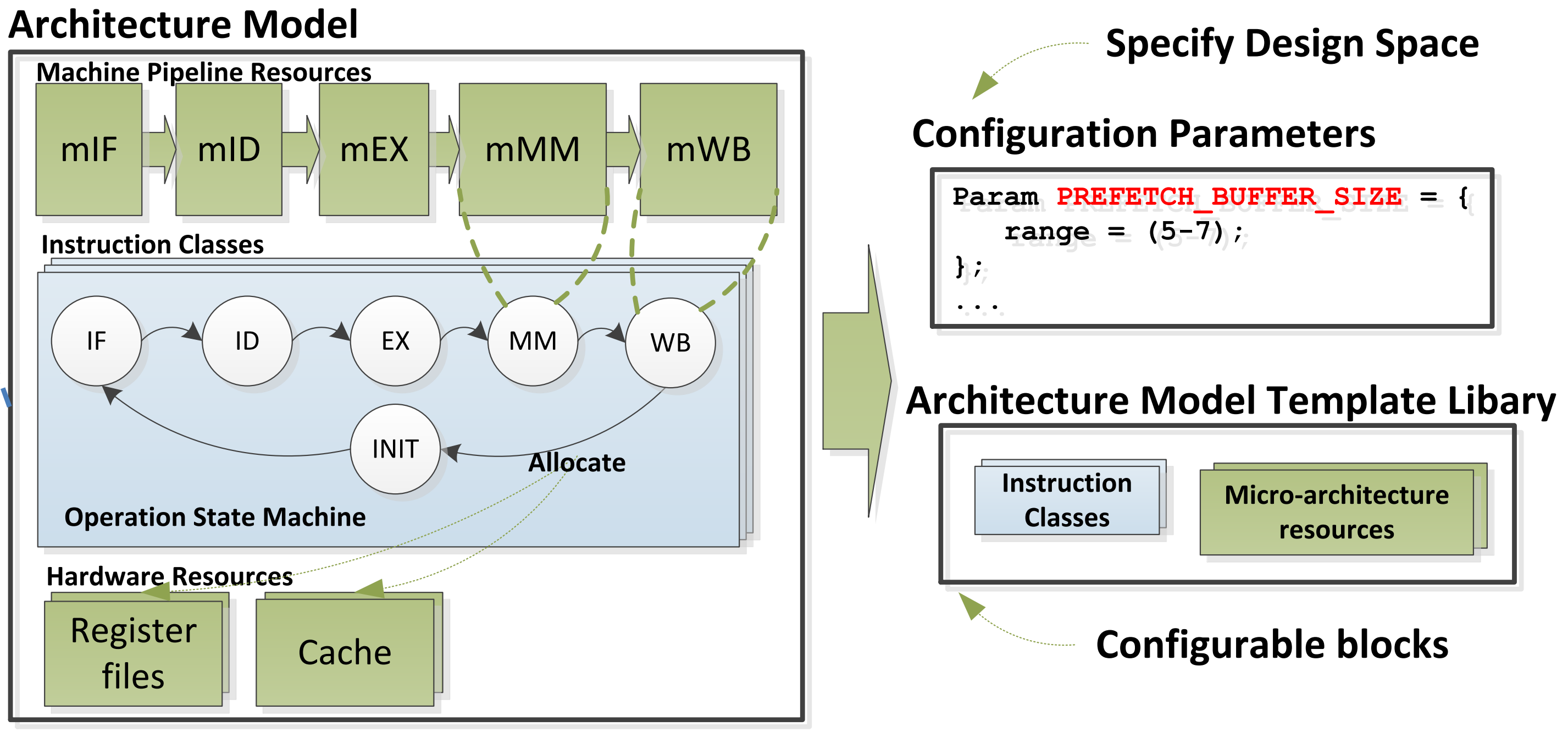
```
param PREFETCH_BUFFER_SIZE = {
  range = (5-7);
};
…
```

Parameterized uADL file

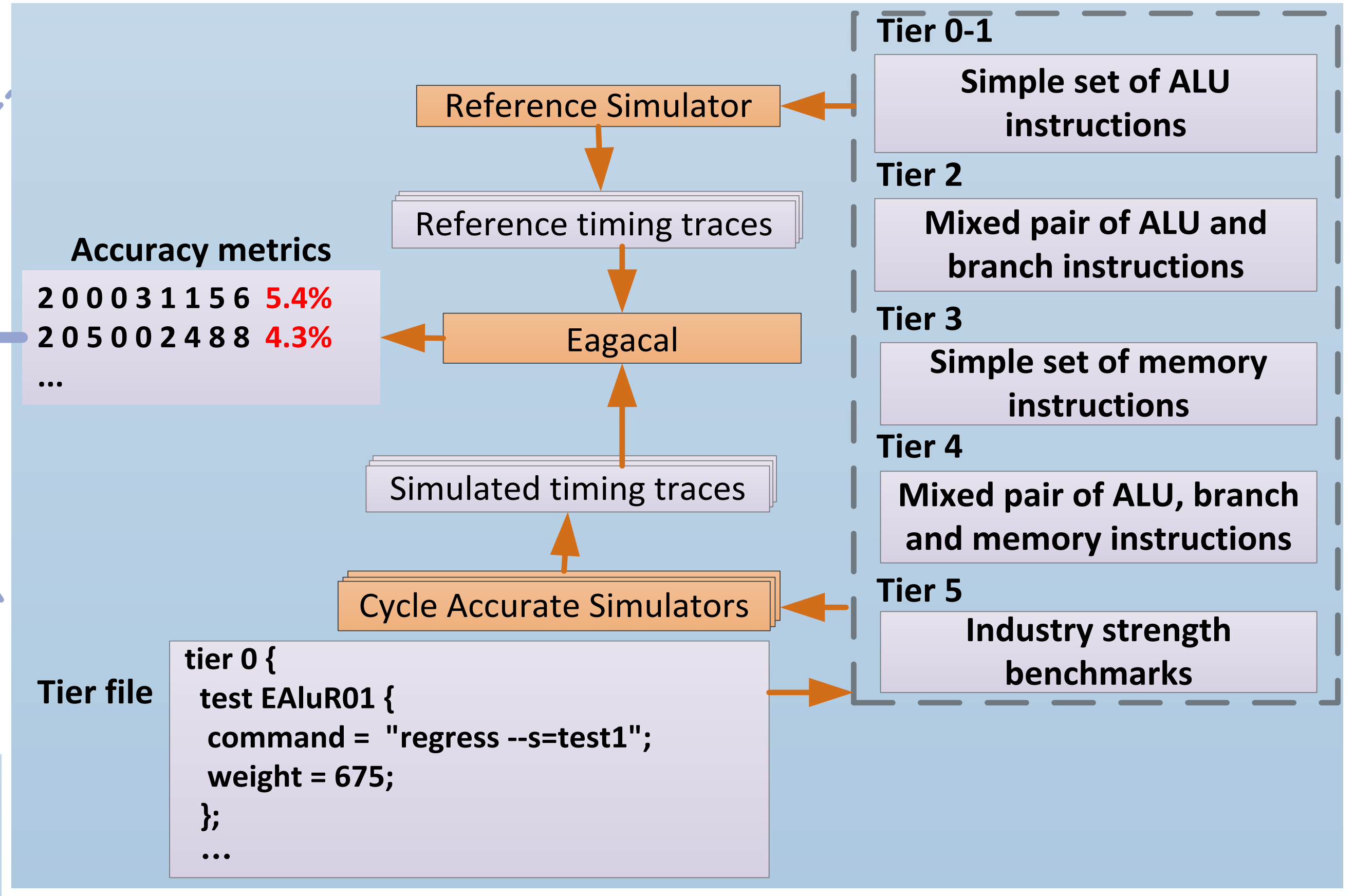
```
define (resources) {
   define (fetchunit Fetcher) {
    fetch_memory = Mem;
    entries =PREFETCH_BUFFER_SIZE;
           … }
…
```

Model Characterization and Parameterization

- ADL/uADL model characterization
  - Generate configurable ADL/uADL model blocks
  - Specify search space through configuration parameters



```
Param PREFETCH_BUFFER_SIZE = {
   range = (5-7);
};
...
```



Reference Simulator
Reference timing traces
Eagacal
Simulated timing traces
Cycle Accurate Simulators
Accuracy metrics
Tier file
Tier 0-1
Simple set of ALU instructions
Tier 2
Mixed pair of ALU and branch instructions
Tier 3
Simple set of memory instructions
Tier 4
Mixed pair of ALU, branch and memory instructions
Tier 5
Industry strength benchmarks

Accuracy metrics

```
2 0 0 0 3 1 1 5 6  5.4%
2 0 5 0 0 2 4 8 8  4.3%
…
```

Tier file

```
tier 0 {
 test EAluR01 {
  command =  "regress --s=test1";
  weight = 675;
 };
 …
```

Successive Multi-Tiered Pruning

- Multiple tiers fitness assignment
  - Quickly prune design space by incrementally adding tiers
  - Introduce a new tier when best individual is not improved
  - Fast fitness assessment of each candidate parameter set

$$Accuracy = 1 - \frac{\sum_{i=1}^{n} W_i \times E_i}{\sum_{i=1}^{n} W_i}$$

$$For\ tier0 - tier4: \quad E_i = \frac{\sum_{j=1}^{W_i} |t_j - t_j'|}{t_{W_i}'}$$

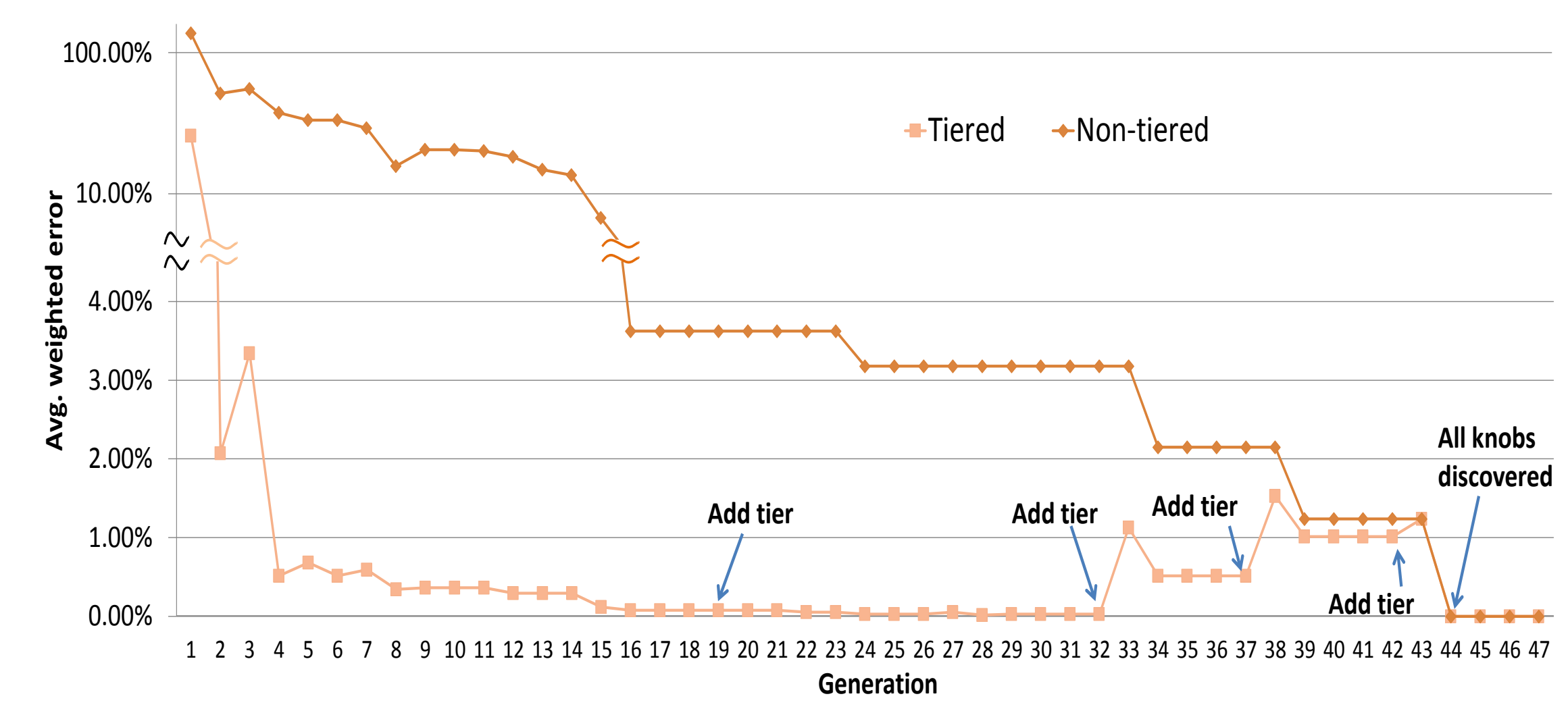
$$For\ tier5: \quad E_i = \frac{|t_{W_i} - t_{W_i}'|}{t_{W_i}'}$$

W_i : instruction count of *benchmark*$_i$
E_i : timing error of *benchmark*$_i$

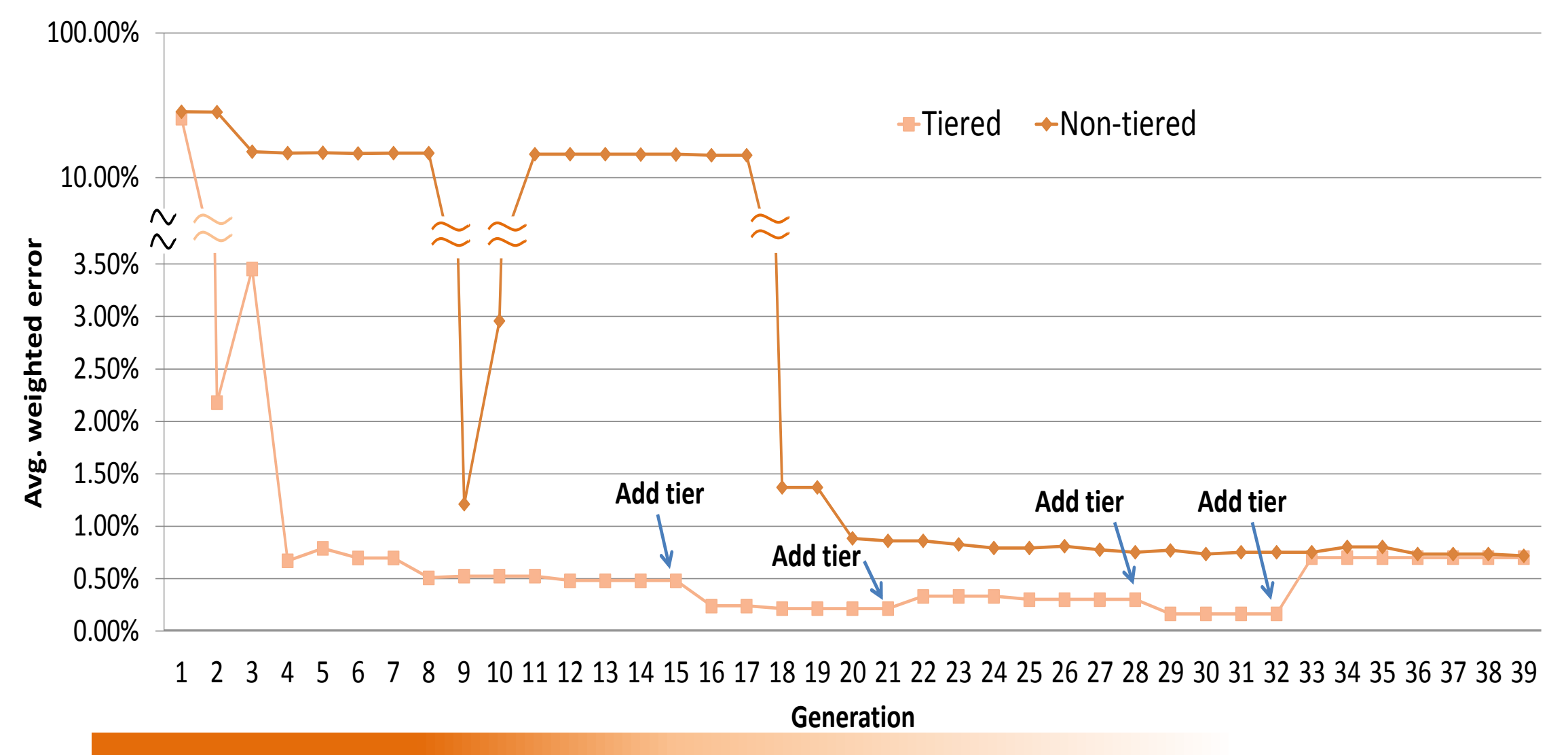
t_j : simulated commit time of *instruction*$_j$
t_j' : reference commit time of *instruction*$_j$

Experimental Results

Calibration Results

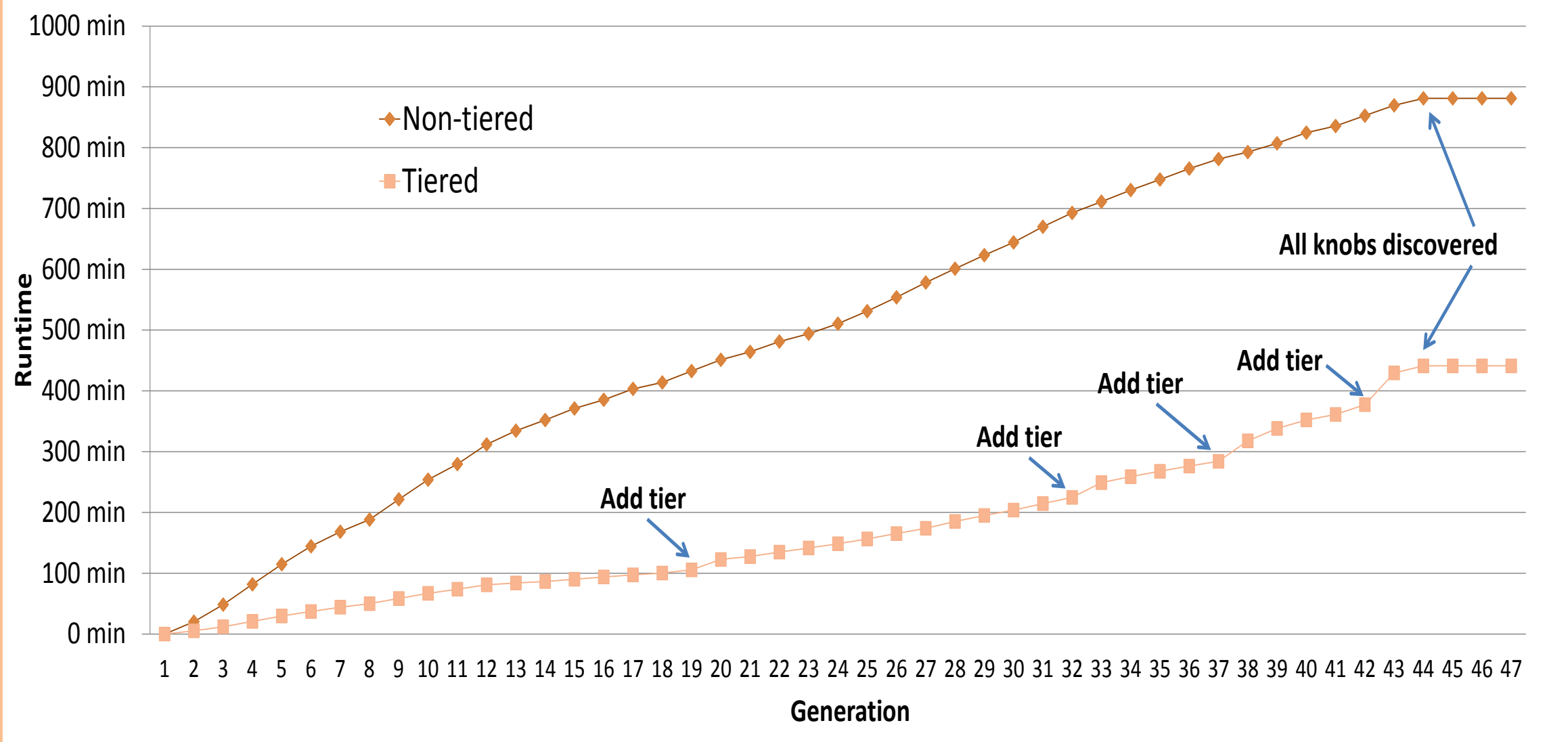


- Automated calibration against reference uADL model
  - uADL model for e200z0 PowerPC core
    - 14 parameters in uAPL file
    - 29 million possible candidates
  - ➢Compare against timing traces generated from reference ADL model



- Automated calibration against RTL reference
  - uADL model for e200z0 PowerPC core
  - ➢Compare against RTL timing reference traces
    - ➢>99% average accuracy across all benchmarks in 8-hour calibration

Calibration Speed



- Calibration runtime
  - Hierarchical benchmarks sets for evaluation and simulation
    - Add tier and re-evaluate when no more improvement of the population
  - ➢Reduce calibration time by 46% compared to non-tiered approach

Technology Transfer

- Reports and publications
  - S. Chakravarty, A. Gerstlauer. "Performance and Power Modeling Case Study", Report, Jan. 2011.
  - S. Chakravarty, A. Gerstlauer. "ADL Driven Back-Annotation and Simulation Methodology for Host Compiled Performance and Power Modeling", Report, July 2011.
  - S. Chakravarty, A. Gerstlauer. "Host Compiled Performance and Power Modeling of Embedded Systems," SRC TECHCON, August 2011.
  - S. Chakravarty, A. Gerstlauer, "Automated, ADL-Driven Generation of Host-Compiled Platform Models," Report, December 2012.
  - S. Chakravarty, Z. Zhao, A. Gerstlauer, "Automated, Retargetable Back-Annotation for Host-Compiled Power and Performance Modeling," *CODES+ISSS*, Montreal, Canada, September 2013.
  - Z. Zhao, S. Chakravarty, A. Gerstlauer, "Automated, Retargetable Back-Annotation for Host-Compiled Performance and Power Modeling," *SRC TECHCON*, Aug. 2013. ***(best in session)***
  - Z. Zhao, G. Morrison, A. Gerstlauer, "Automated ADL Model Calibration and Tuning," Report, Dec. 2013.
- Industrial interactions
  - Leveraging Freescale's open-source ADL technology
    - http://opensource.freescale.com/fsl-oss-projects/
  - Regular (bi-weekly) project discussions (Freescale, TI)
  - Internships at Freescale (6/2013-8/2013)
    - Automated discovery of ADL based cycle-accurate reference models