# The Fuzzy Correlation between Code and Performance Predictability

Murali Annavaram, Ryan Rakvic, Marzia Polito, Jean-Yves Bouguet, Richard Hankins, Bob Davies

Intel Corporation

Acknowledgements to

John Shen, Carole Dulong, Brad Calder

# *Why Correlate Code and Performance?*

- Predict CPI by observing *just* EIPs (or PCs)
  - *Assume*, similar EIP sequence -> similar CPI
  - Shown to work well for some CPU2K benchmarks
  - For improving simulation speed and dynamic optimizations
- Do server workloads exhibit this correlation?
  - Large code base and non-loopy code path
  - Processes communicate through inter process communication
  - Use of OS services
- Regression Trees
  - Quantify CPI prediction accuracy using EIPs
  - Find upper bound for correlation

# Workloads & Experimental Infrastructure

- Three representative server workloads
  - ODB-C
    - OLTP benchmark on Oracle 10g RDBMS
  - ODB-H
    - DSS benchmark on Oracle 10g RDBMS
  - SPECjAppServer (SjAS)
    - 3-Tier Application
    - Focus on middleware application server running BEA Weblogic
- Workloads tuned for maximum CPU utilization
- Hardware Configuration
  - Itanium-2 processor based system
  - Red Hat 2.1 + kernel 2.4.9-e.10smp
  - 16 GB of DDR memory
  - 34 Ultra320 SCSI 73 GB drives (used in ODB-C and ODB-H)
  - 200 MHz FS Bus

# Tool Chain: Step 1: Data Collection


- VTUNE: Non-intrusive performance monitoring of physical systems
- Samples hardware counters
- No code instrumentation/recompilation
- Collects EIP & TSC once every 1M instructions
  - 2% execution overhead
  - Sampling at 100K instructions has negligible effect
- Sampled EIPs are a good approximation for code path

# Step 2: Vector Creation

EIP and TSC

**Vector Creation**

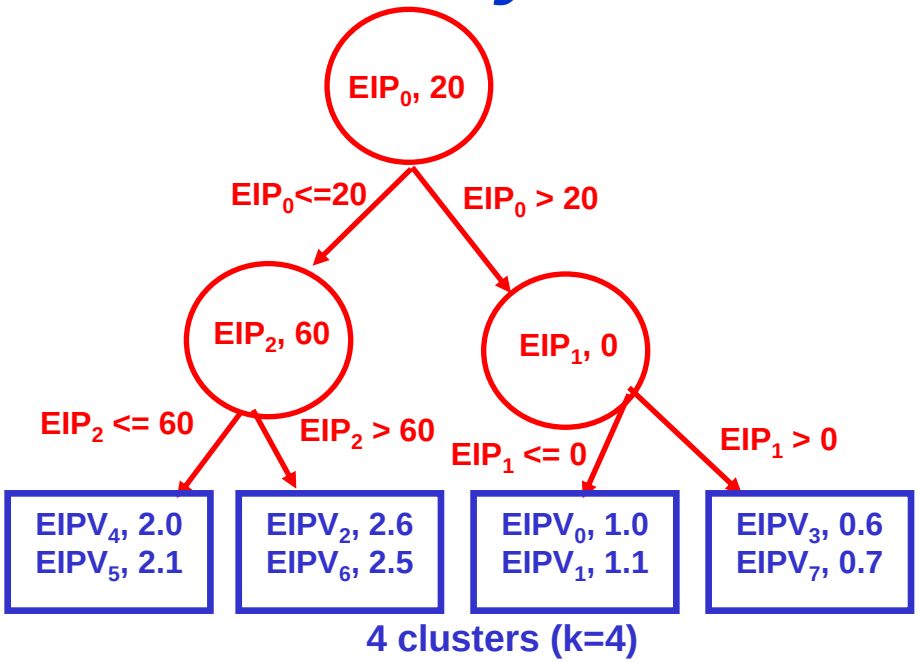
| | $EIP_0$ | $EIP_1$ | CPI |
|---|---|---|---|
| $EIPV_0$ | 100 | 0 | 1.0 |
| $EIPV_1$ | 20 | 80 | 2.0 |
| $EIPV_2$ | 100 | 0 | 1.1 |

(EIPV, CPI)

- Execution divided into 100M instruction interval
- Create 1 EIPV per interval
  - 100 VTune samples per EIPV
  - EIPV has sample count of each unique EIP in that interval
  - Any EIP not sampled in an interval has zero count
- Instantaneous CPI associated with EIPVs
  - CPI = (End Time stamp - Begin Time Stamp) / 100M

# Step 3: Regression Tree Analysis

| | $EIP_0$ | $EIP_1$ | $EIP_2$ | CPI |
|---|---|---|---|---|
| $EIPV_0$ | 100 | 0 | 0 | 1.0 |
| $EIPV_1$ | 80 | 0 | 20 | 1.1 |
| $EIPV_2$ | 0 | 20 | 80 | 2.6 |
| $EIPV_3$ | 80 | 20 | 0 | 0.6 |
| $EIPV_4$ | 20 | 20 | 60 | 2.0 |
| $EIPV_5$ | 20 | 20 | 60 | 2.1 |
| $EIPV_6$ | 20 | 0 | 80 | 2.5 |
| $EIPV_7$ | 80 | 20 | 0 | 0.7 |

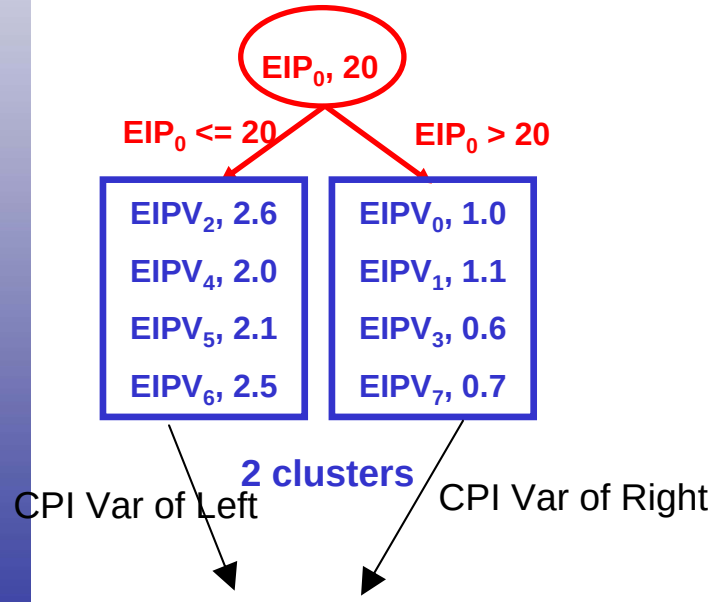



- Divide EIPVs into clusters where sum of CPI variance minimized
  - CPI optimally drives EIPV clustering; machine dependent
  - *By construction*, EIPVs in the same cluster will have similar CPI
- Example: The CPI variance of regression tree clusters is smallest for all possible clusters of size 4

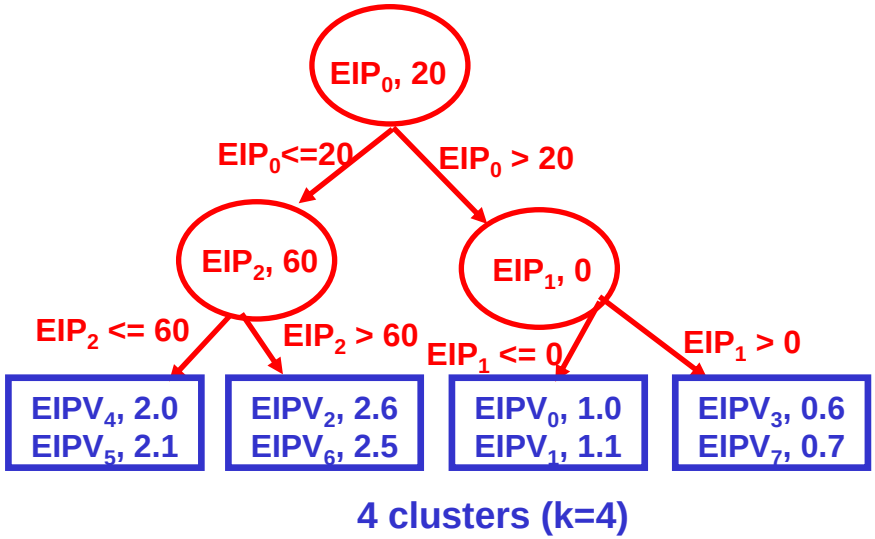
- K-means comparison: Clustering using distance between vectors
  - Does *not* use CPI values in clustering; machine independent

# Computing Relative Error Metric

- In our study, limit the tree size to 50 clusters
  - > 50 clusters does not reduce CPI variance
- Compute CPI variance for all K clusters for each tree $T_K$ (1<=K<=50)
- Relative Error (RE) = weighted sum of cluster CPI variance / overall CPI variance




RE=(.087+.057)/.662=0.22

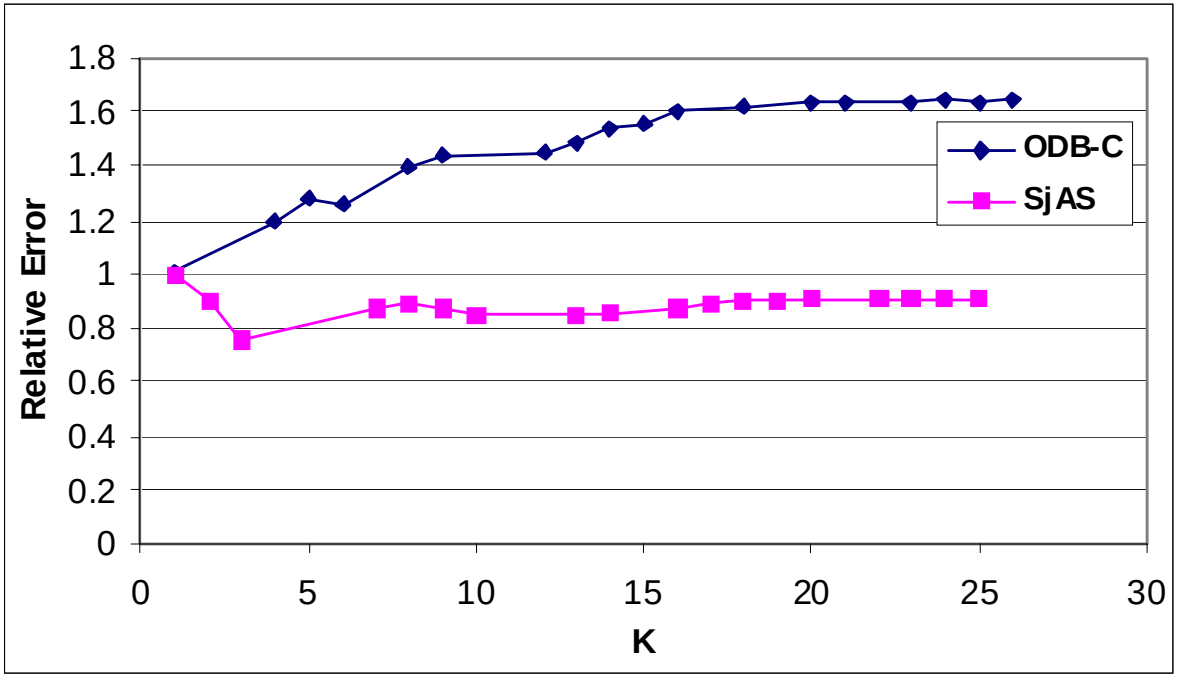



RE=(.005+.005+.005+.005)/0.662=0.03

# *Interpreting Relative Error Metric*

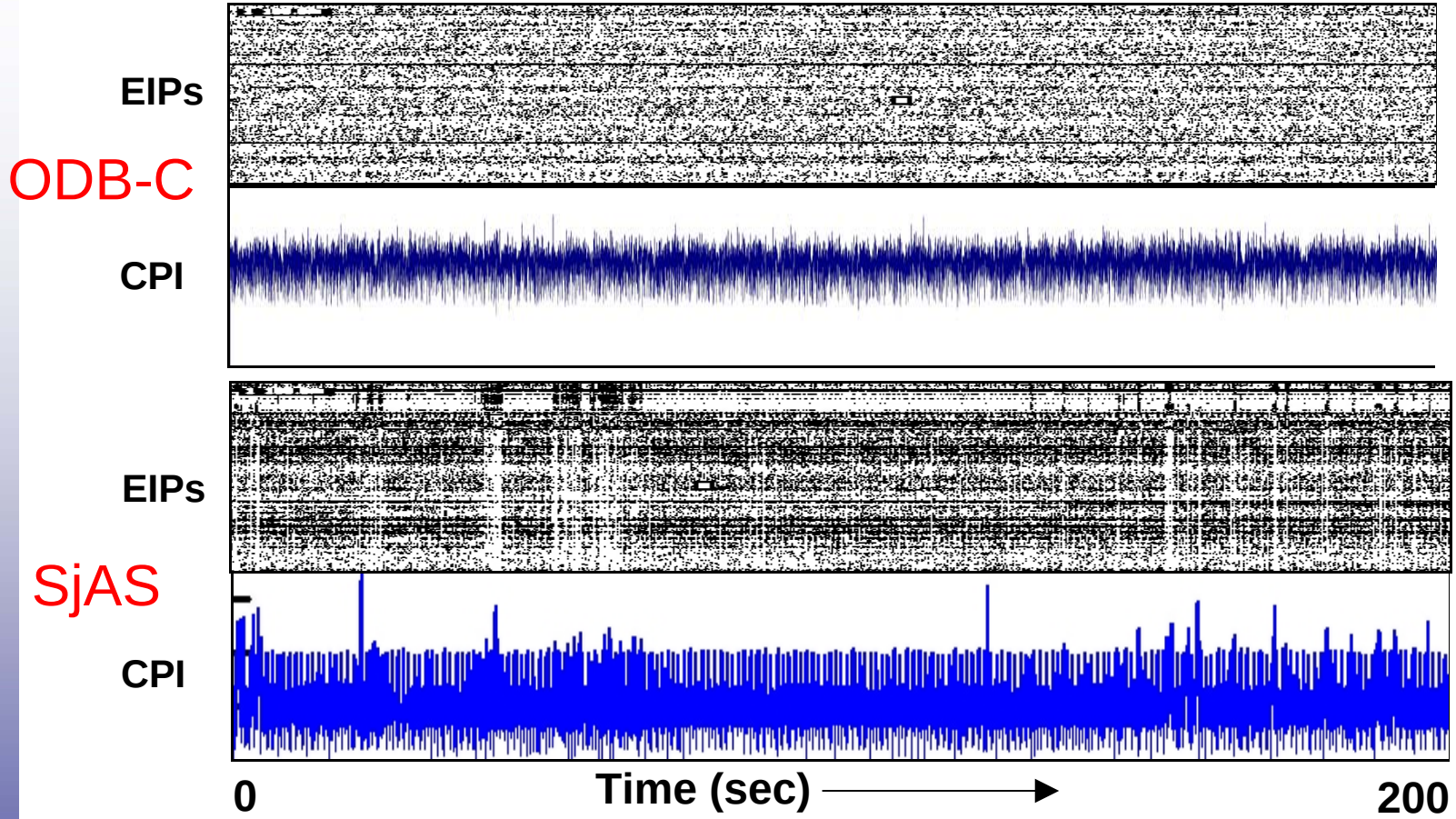
- RE represents CPI variance explained by EIPVs
  - RE=0.15 means that 85% of the CPI variance *explained* by EIPVs.
  - If RE~1 then EIPVs have no relationship with CPI
- Small RE + Small tree size (K)
  - workload behavior exhibits a small number of dominant phases
- If regression tree is large
  - Irrespective of RE, EIPVs and CPI relationship not regulated by few dominant phases
- If CPI variance is small (uniform CPI)
  - No need for regression trees
  - Simple average is acceptable

# Regression Tree Results - ODB-C and SjAS



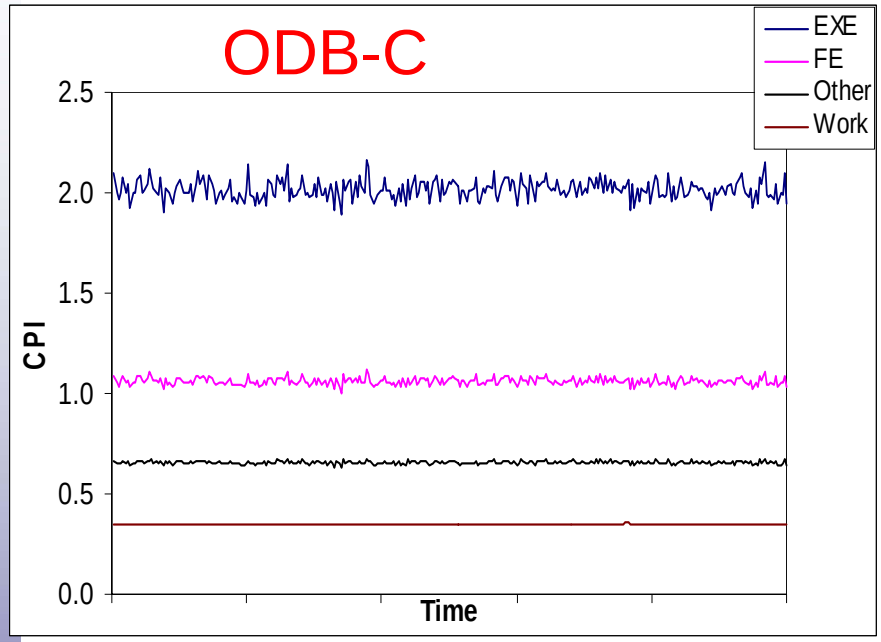

- ODB-C
  - CPI has no correlation with EIPs
- SjAS
  - Only 20% of CPI variance explained by EIPs

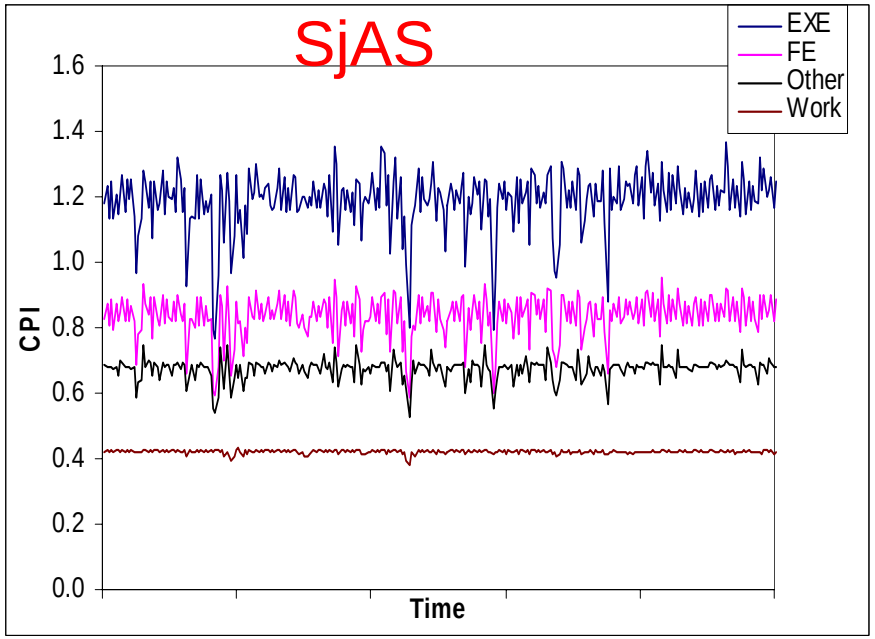
# A Visual Explanation - ODB-C and SjAS




- Many unique EIPs compared to SPEC
  - 24K in ODB-C, 31K in SjAS, compared to 646 in MCF from CPU2K
- Small CPI variance
  - 0.01 in ODB-C and 0.03 in SjAS
- Performance independent of EIPs

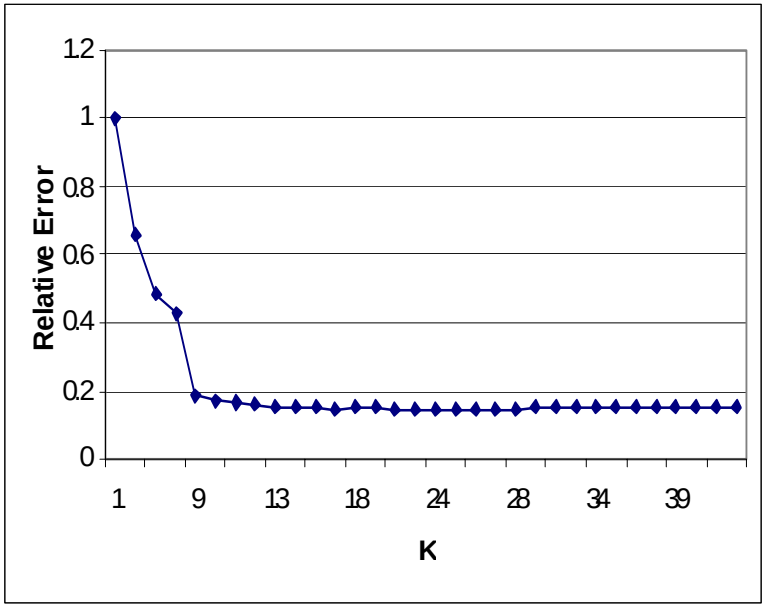
# CPI breakdown – ODB-C and SjAS



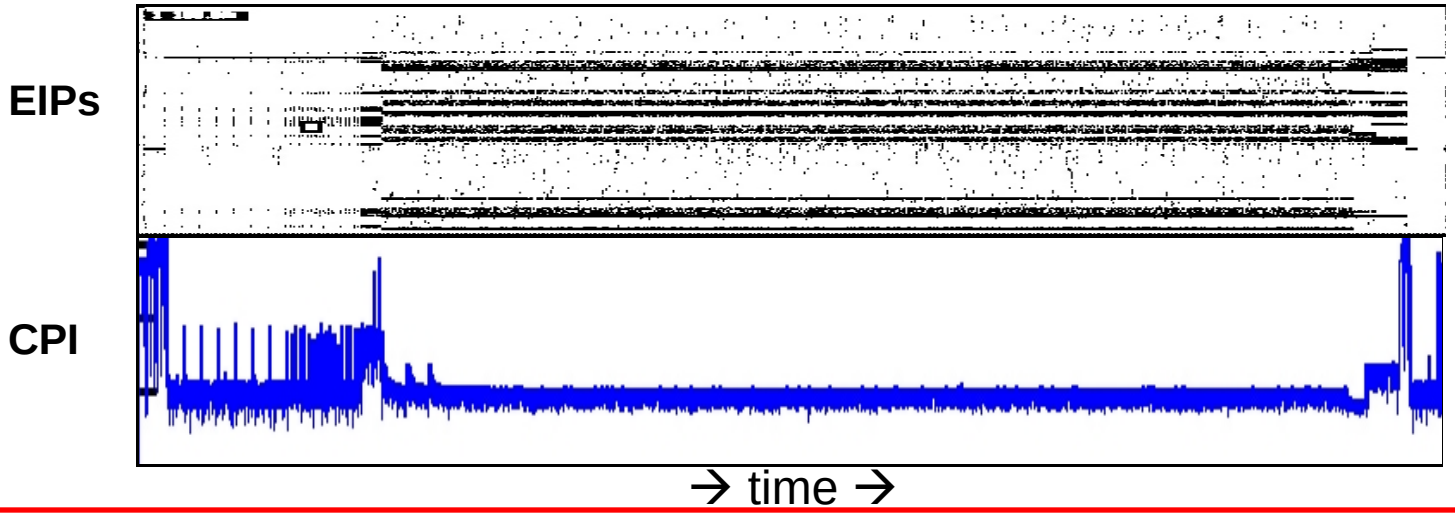




- L3 misses occur frequently and uniformly
  - 50% of ODB-C CPI, 35% of SjAS CPI due to L3
- L3 misses overwhelmed other microarchitectural bottlenecks
  - CPI determined by L3 miss latency

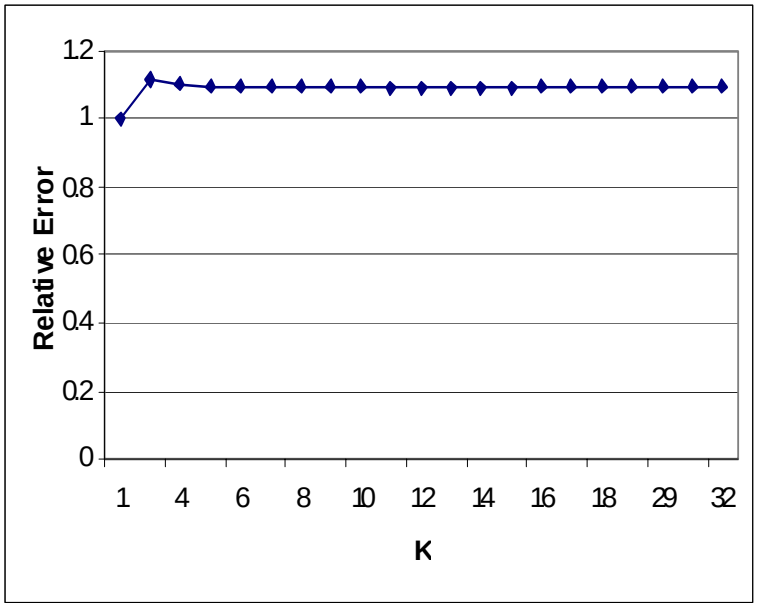
# ODB-H – Q13



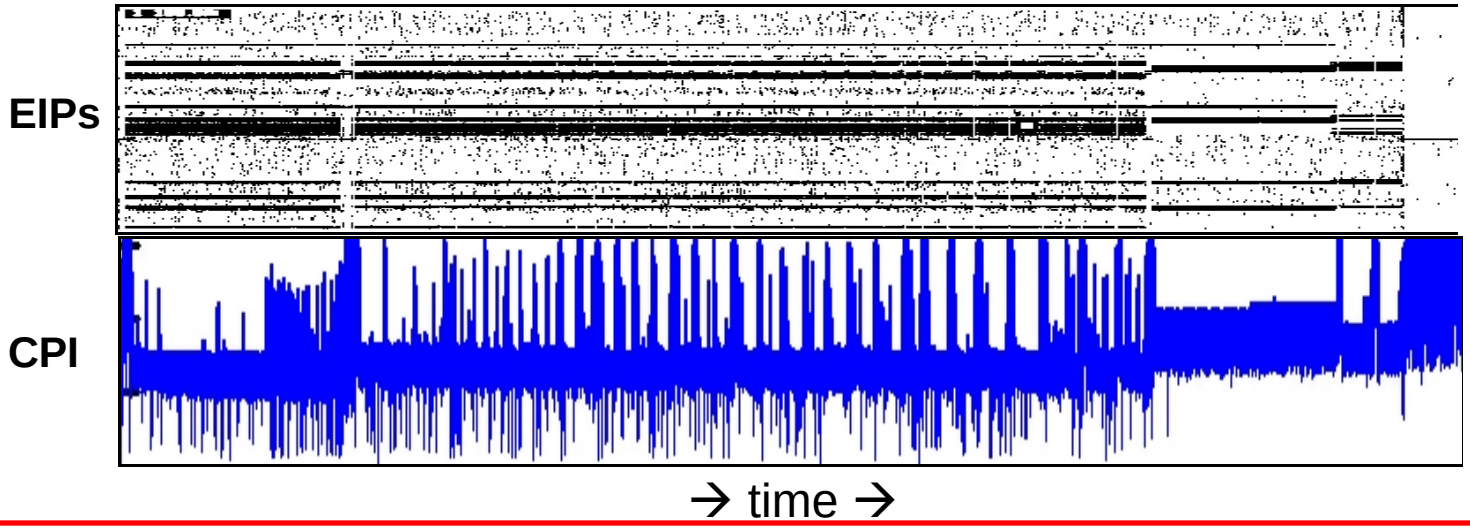

- Three functions
  - *Sequential scan*, join, sort
- Repeated execution on large input data
- Relative error drops to ~0.15
  - 85% of CPI variance explained by EIPs
- Distinct phases



# ODB-H – Q18




- Same 3 functions
  - but uses *index scan* instead of sequential scan
  - more cache and branch misses
- CPI varies widely for the same code
- Relative error is 1.1



# Quadrant Classification

Relative Error ↑

| Bmark | CPI Var | RE | Bmark | CPI Var | RE |
|---|---|---|---|---|---|
| ODB-C | 0.00 | 1.01 | Q18 | 0.65 | 1.00 |
| Q1 | 0.01 | 0.78 | SjAS | 0.03 | 0.90 |
| Q19 | 0.01 | 0.37 | Q7 | 0.04 | 0.19 |
| Q3 | 0.01 | 0.36 | Q5 | 0.03 | 0.18 |
| Q16 | 0.01 | 0.25 | Q10 | 0.02 | 0.18 |
| | QI | | Q15 | 0.03 | 0.18 |
| | | | Q9 | 0.02 | 0.17 |
| | | | Q4 | 0.02 | 0.16 |
| Q17 | 0.01 | 0.15 | Q22 | 0.04 | 0.15 |
| Q2 | 0.01 | 0.09 | Q21 | 0.02 | 0.15 |
| | | | Q13 | 0.02 | 0.15 |
| | QII | | Q8 | 0.02 | 0.12 |
| | | | Q20 | 0.03 | 0.12 |
| | | | Q14 | 0.08 | 0.09 |
| | | | Q12 | 0.06 | 0.06 |
| | | | Q6 | 0.03 | 0.06 |
| | | | Q11 | 0.02 | 0.04 |

QIII

QIV

CPI Variance →

- Classify benchmarks into four quadrants using CPI and RE
- Q-I and Q-II have low CPI variance
  - Relative Error is irrelevant
  - Uniform sampling is OK
- Q-III : Predicting CPI from EIPVs alone can *not* achieve accuracy
  - Machine dependent parameters needed to capture CPI variations
- Q-IV benchmarks benefit from simple EIP based phase detection

# Summary

- Using regression trees to identify optimal EIP-CPI relationship in three server workloads
- ODB-C and SjAS
  - CPI has no correlation with EIPs
  - Large code segments
  - Uniform CPI dominated by L3 misses
- ODB-H exhibit a range of behaviors
  - Small code path
  - Algorithmic changes significantly impact CPI and EIP relation
- Quadrant based classification
  - No single sampling technique effective for all
  - Shows best-suited sampling to accurately capture CPI variance