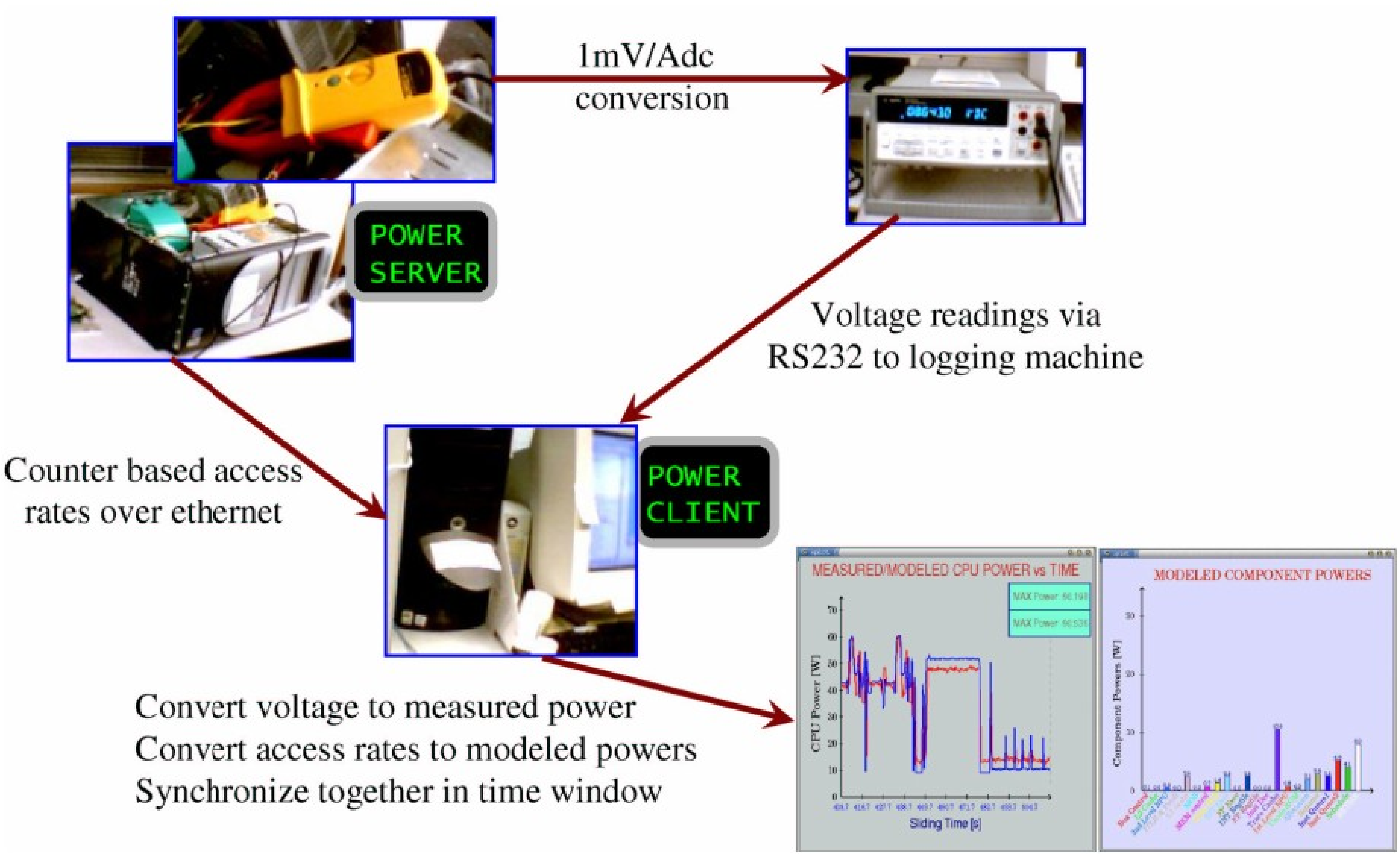
Identifying Program Power Phase Behavior Using Power Vectors

Canturk Isci and Margaret Martonosi
Princeton University

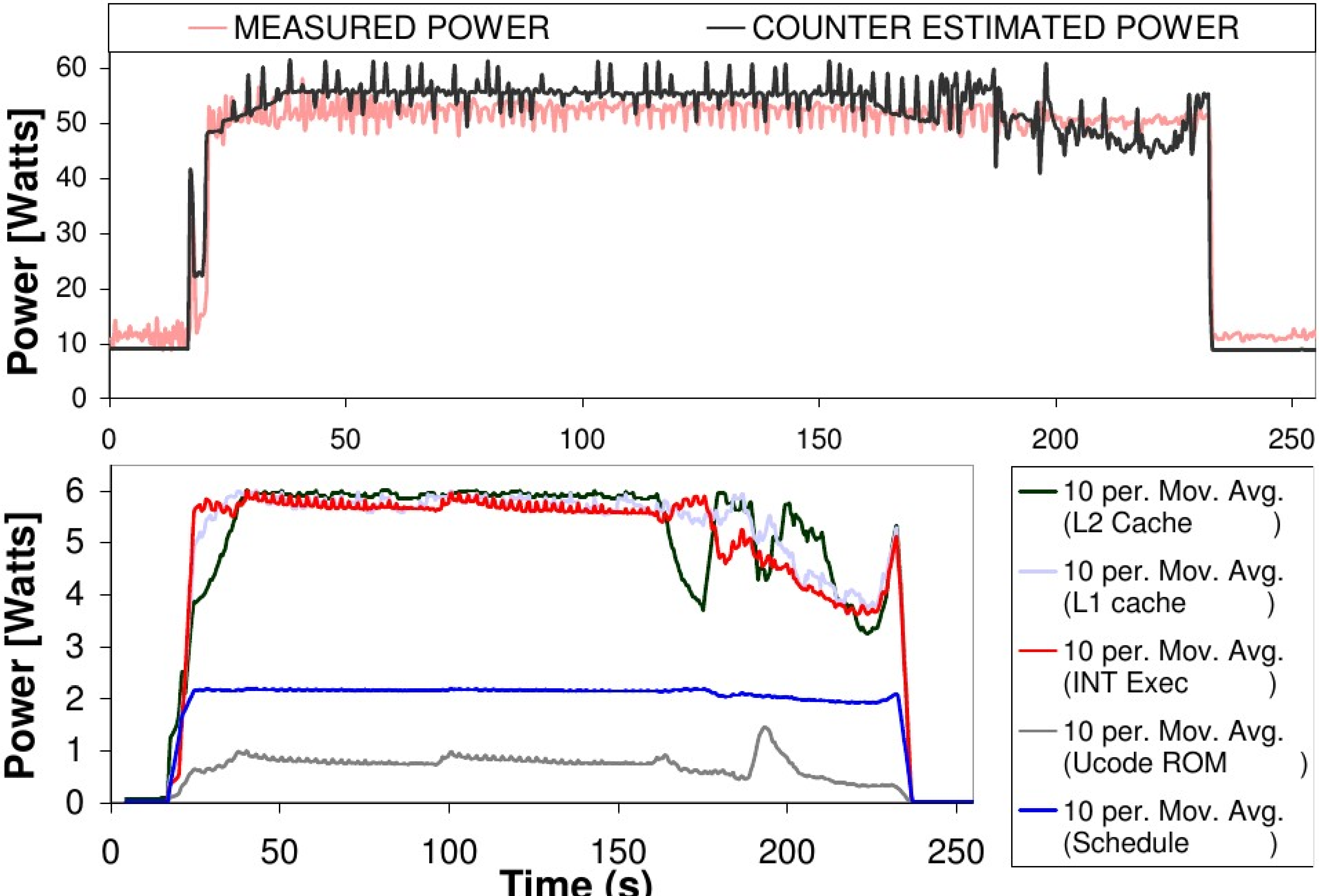
Overview

- A phase analysis method related to power
- Estimates for power values of 22 processor components
- Power vectors acquired at runtime
- Power signatures of programs
- Representative execution points usable for simulation

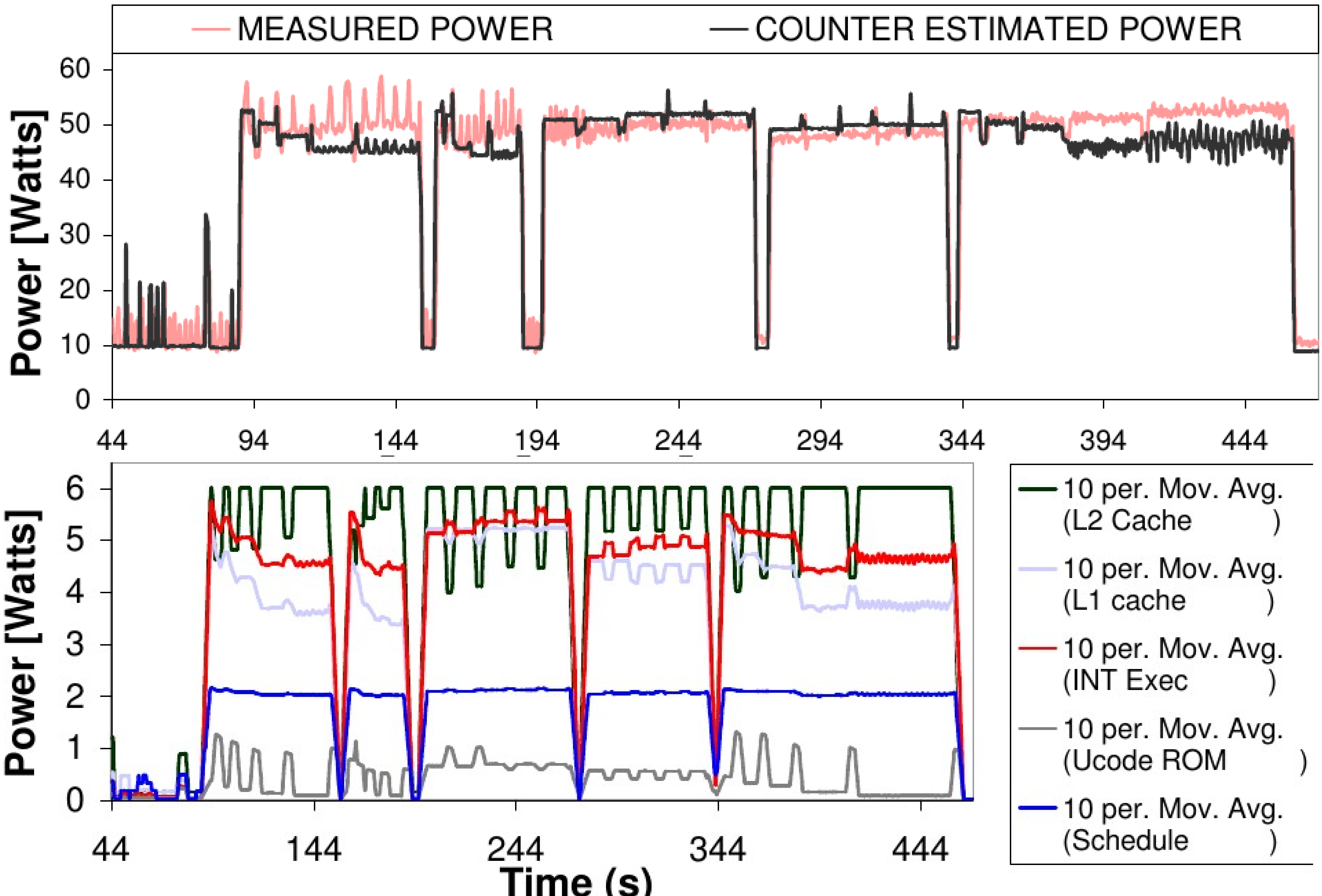
Methodology



Observations (gap)



Observations (gzip)

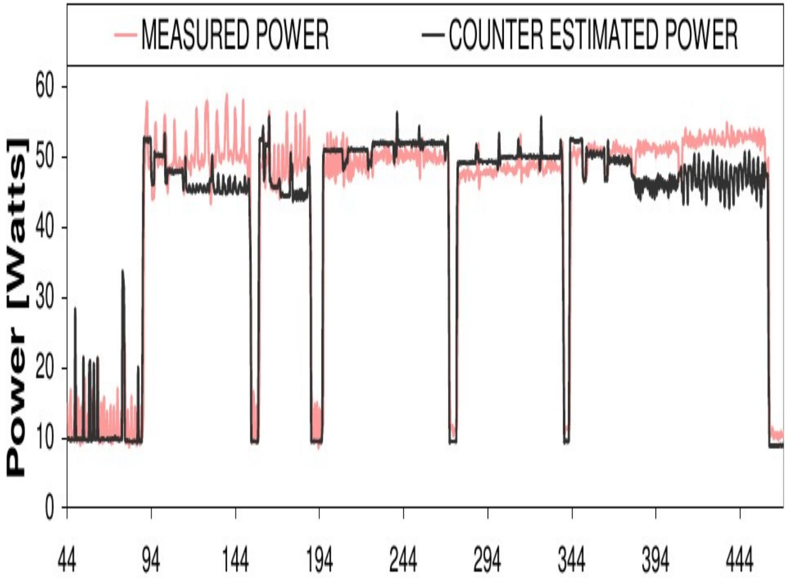


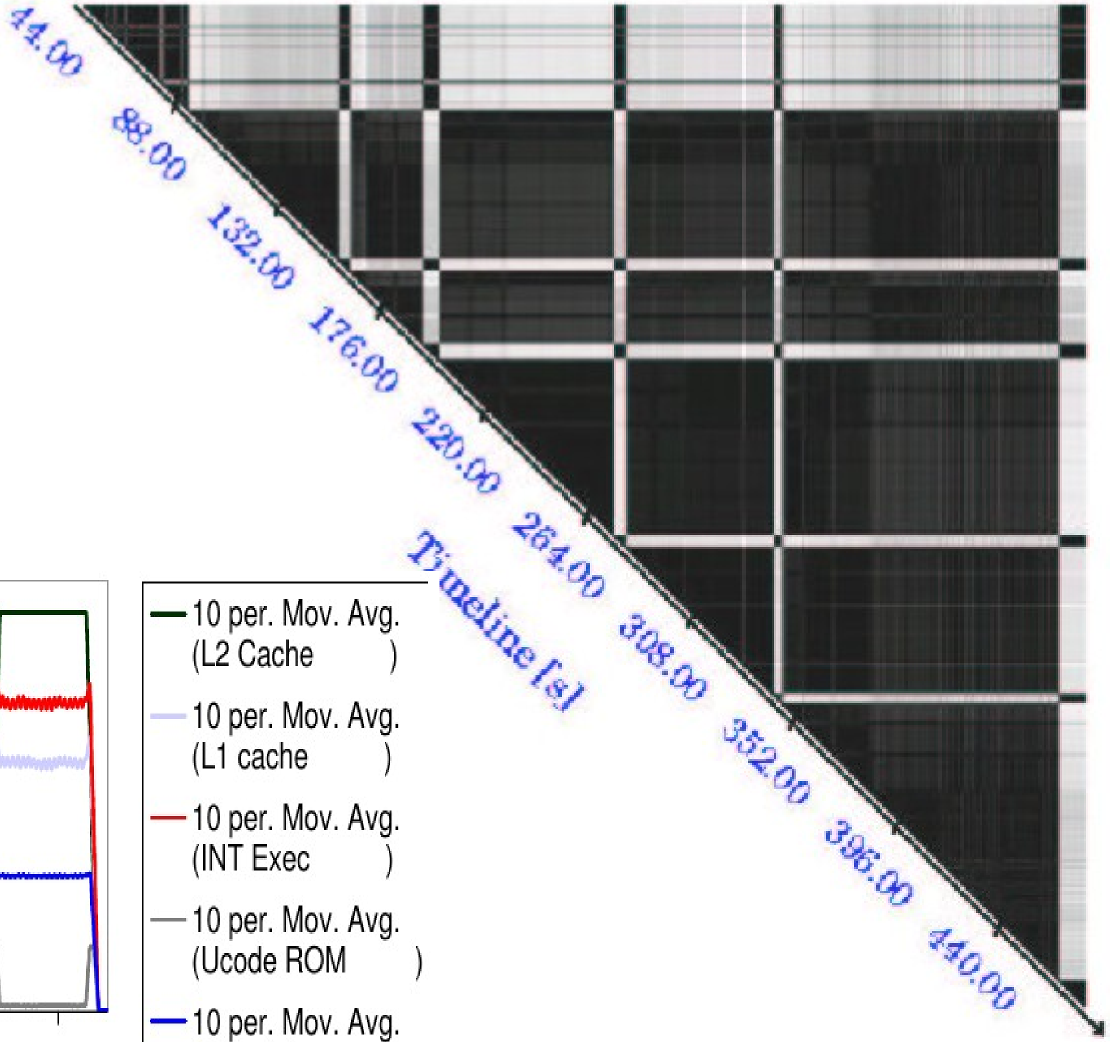
Power vectors for similarity

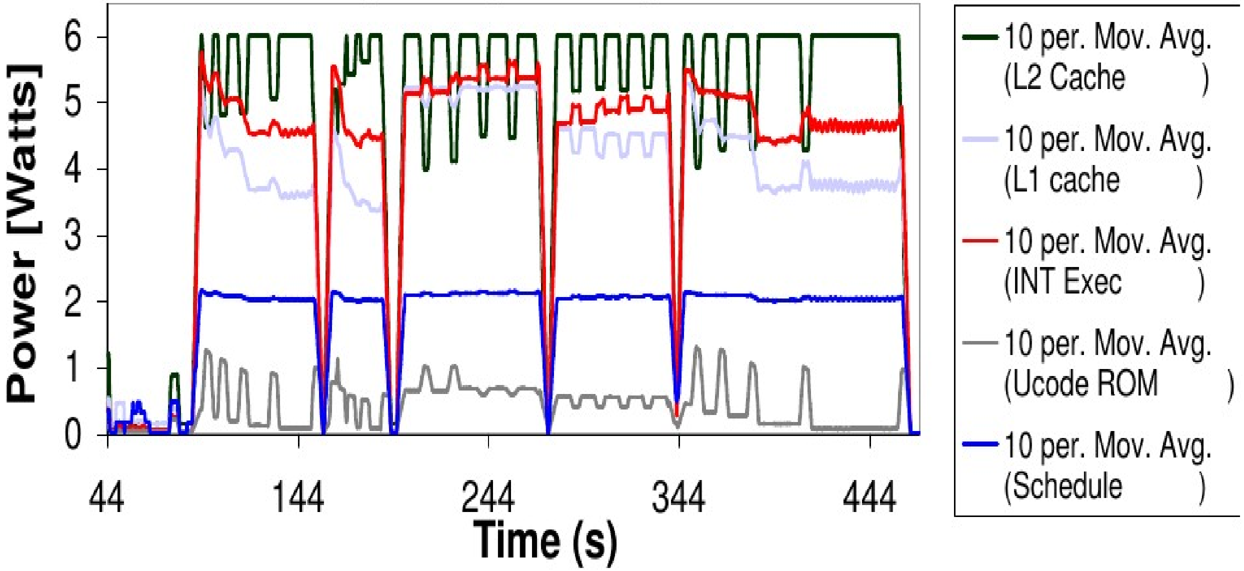
- Spatial closeness of points specified by corresponding power vectors
- Manhattan distance as measure of closeness
- 0 = perfect similarity (black)
- Diagonal = time

First try: based on total power

$$Total\ Similarity\ Matrix(r, c) = |Total\ Power_r - Total\ Power_c|$$







Developing a better metric

- Base similarity on the (original) power vectors
- Manhattan distance between vectors
- $$Original\ Similarity\ Matrix(r,c) = \sum_{i=1}^{22} |PV_r(i) - PV_c(i)|$$
- Problem:
  - vectors of smaller magnitude are bound to be considered similar even though they may point to very different directions in power space.

Developing a better metric (2)

- Normalize vectors to avoid this pitfall

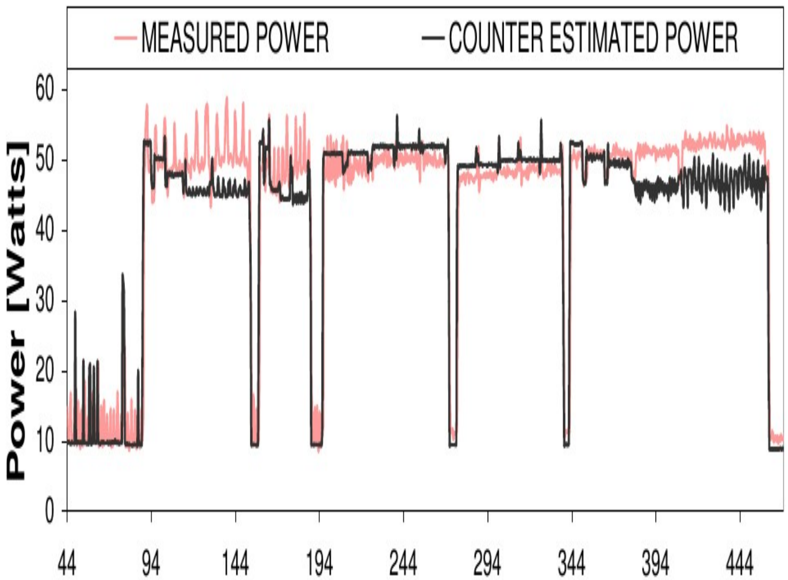
$$Normalized\ Similarity\ Matrix(r,c) = \sum_{i=1}^{22} |NPV_r(i) - NPV_c(i)|$$

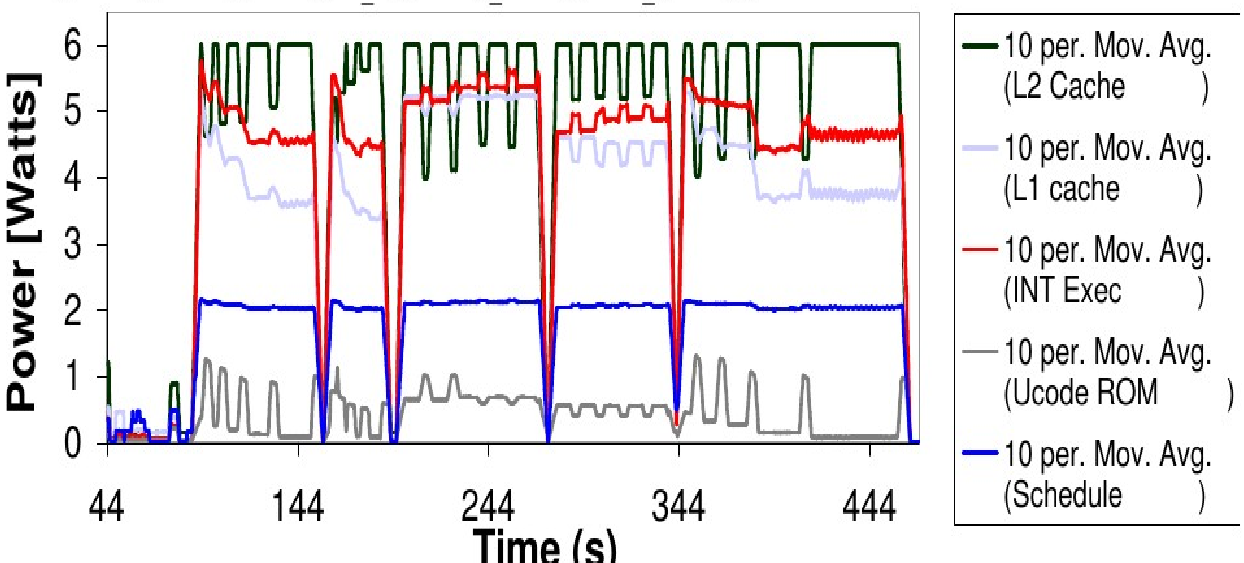
- But: indifferent to magnitude of vectors as ratios prove to be similar

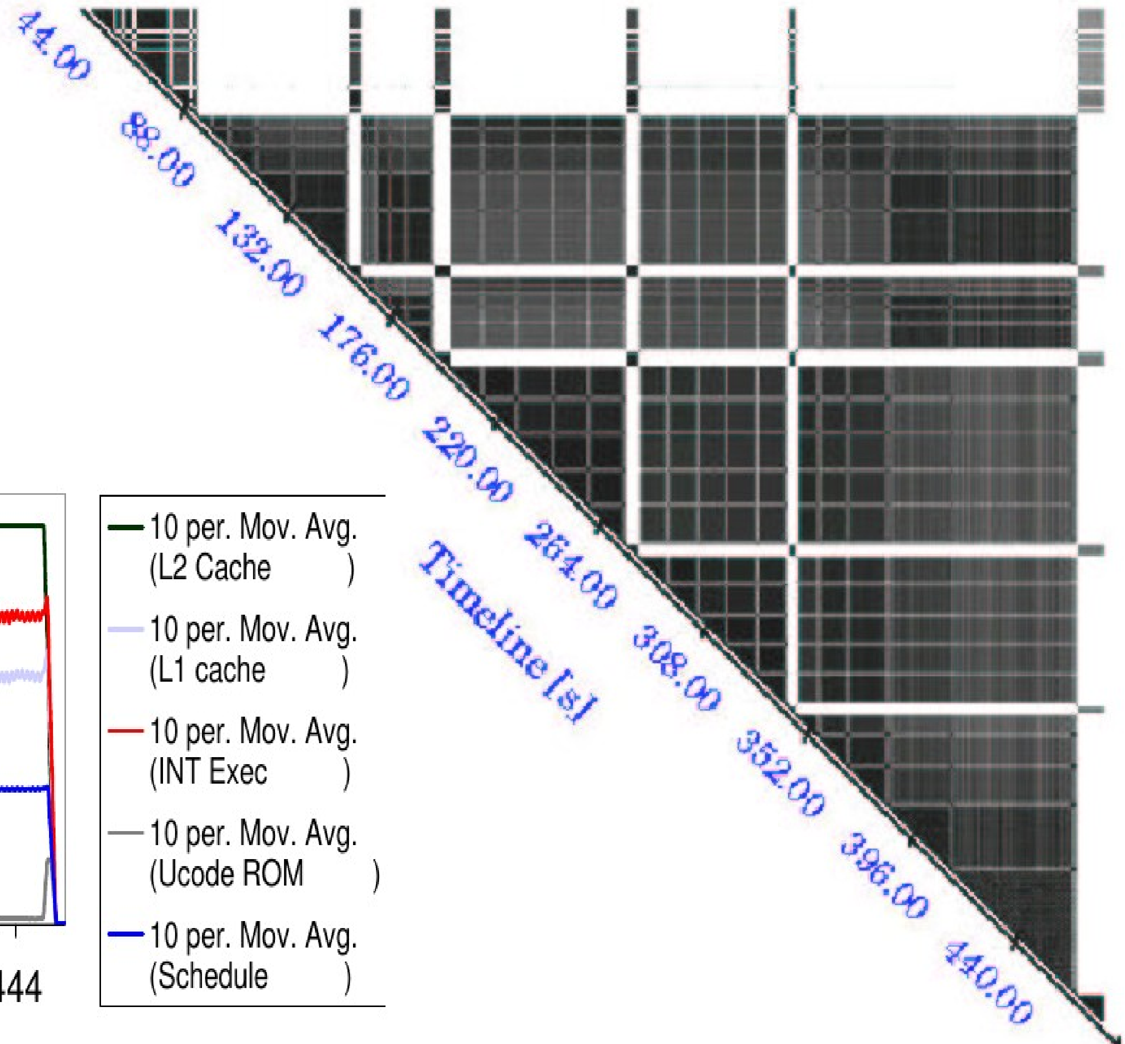
→ combine both metrics!

The final metric

$$FM(r,c) = \min\left(\frac{OM(r,c)}{\max_{r',c'}(OM(r',c'))} + \frac{NM(r,c)}{\max_{r',c'}(NM(r',c'))}, 1 \right)$$







Similarity Groups

- Aim:

  reduced workload size for benchmarks

  still capture most of its power behavior

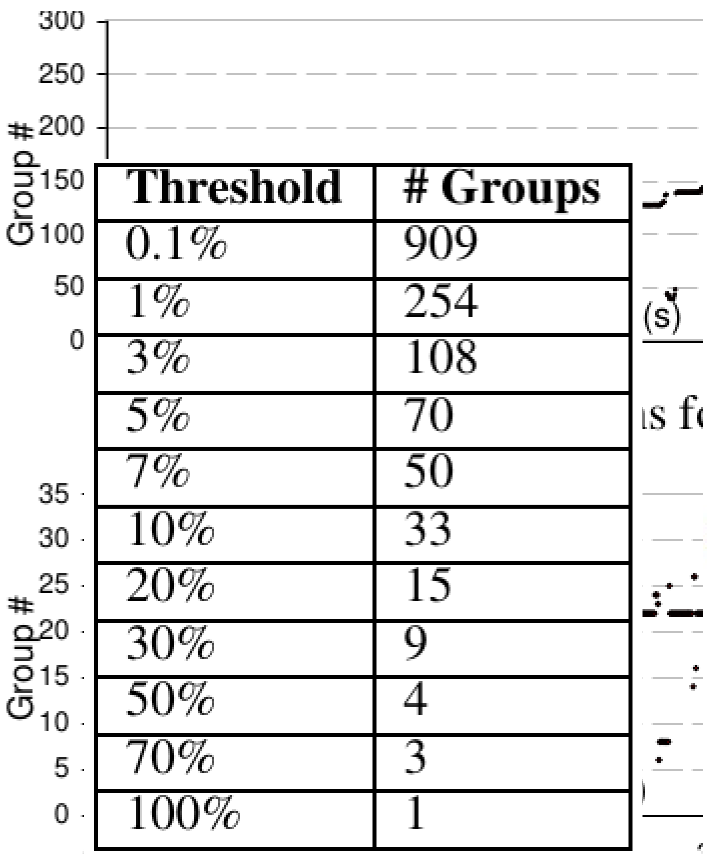
  power signature using representative vectors

- Goals for Thresholding:

  (1) Grouping execution points—power vectors—based on their similarity

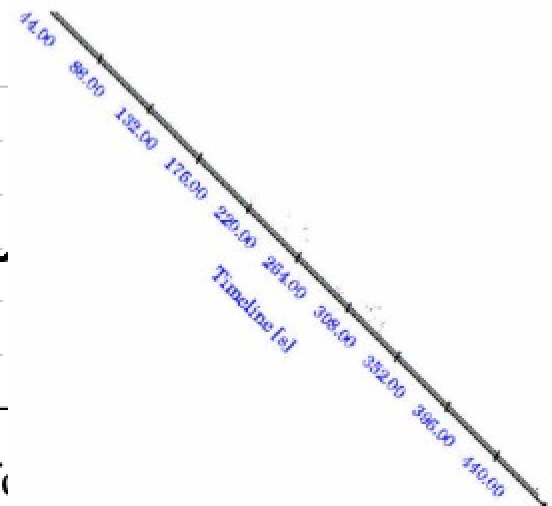
  (2) Representing power behavior with reasonable accuracy with a small number of “signature vectors”

Thresholding

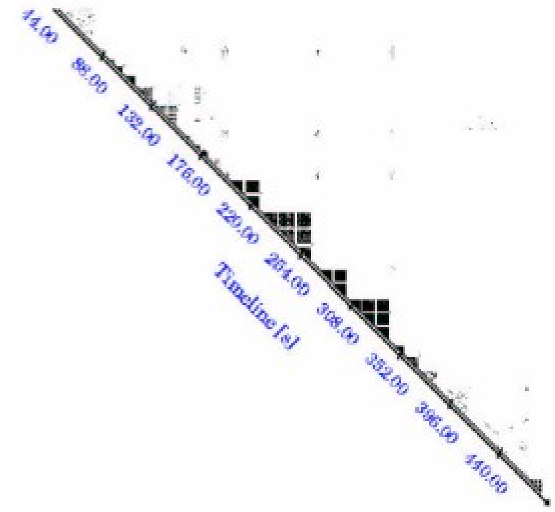


Threshold	# Groups
0.1%	909
1%	254
3%	108
5%	70
7%	50
10%	33
20%	15
30%	9
50%	4
70%	3
100%	1

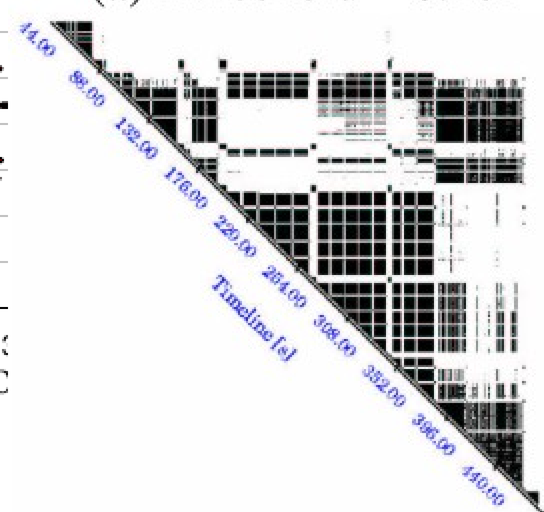
(b) Gzip group distributions fc



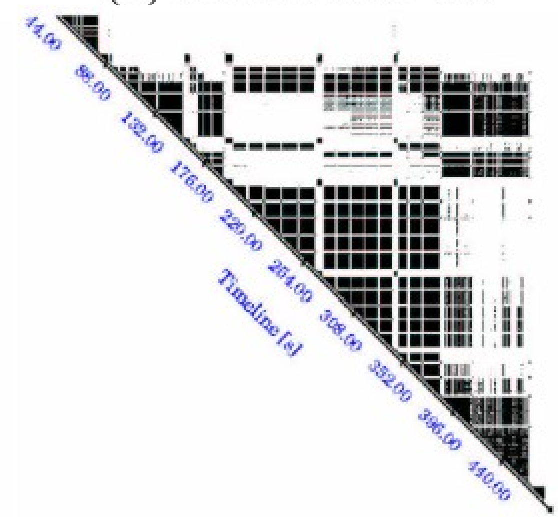
(a) Threshold = 0.1%



(b) Threshold = 1%

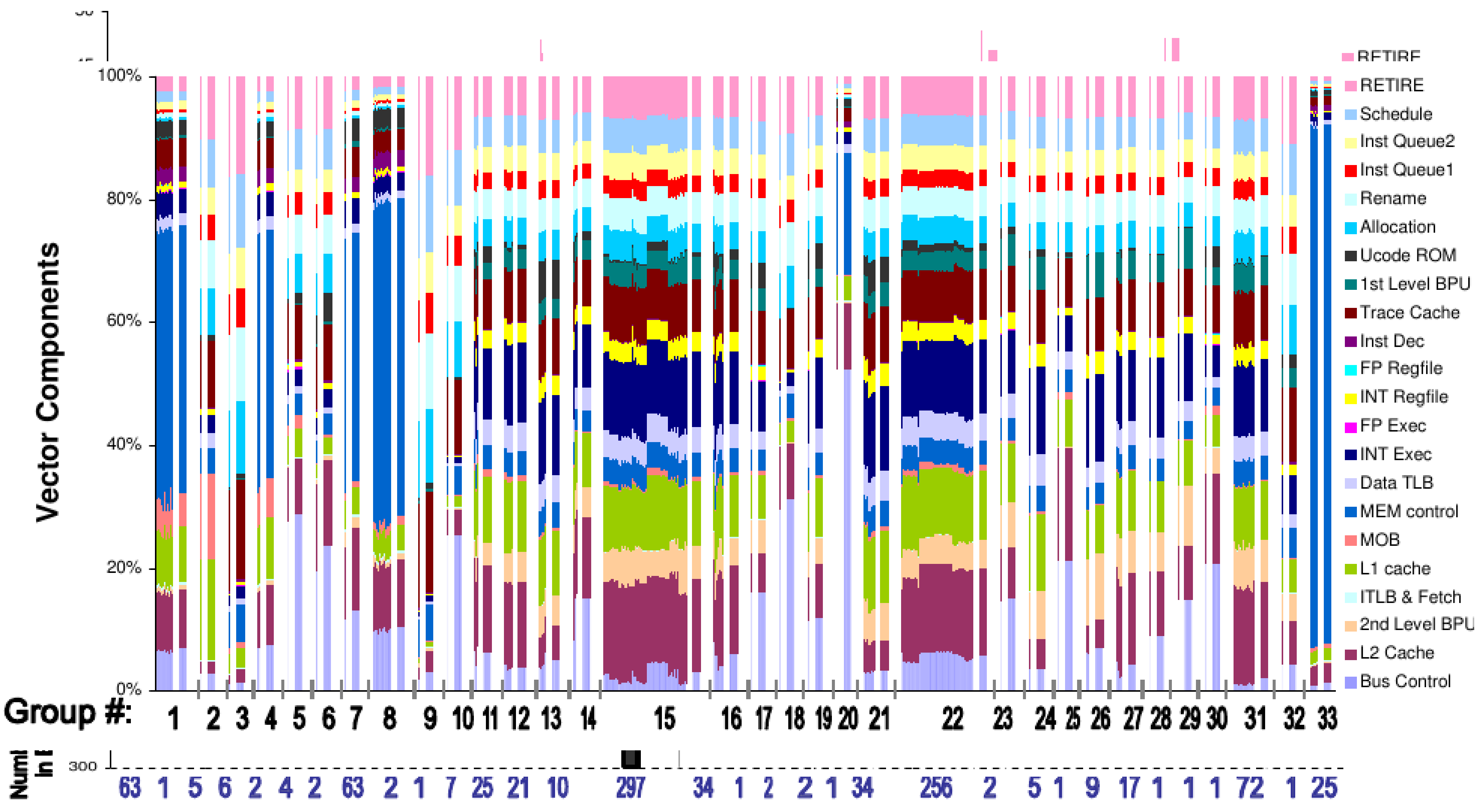


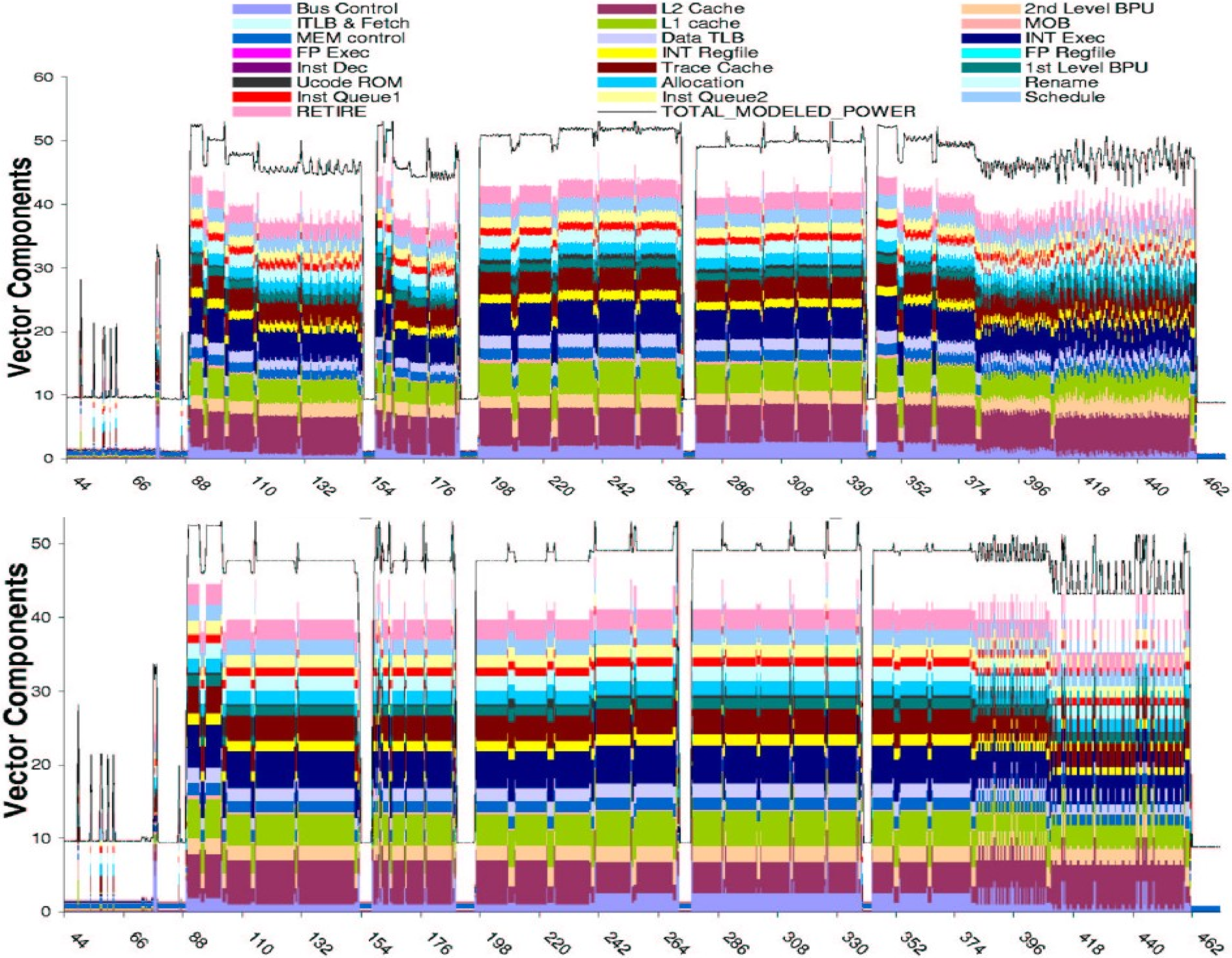
(c) Threshold = 10%



(c) Threshold = 10%

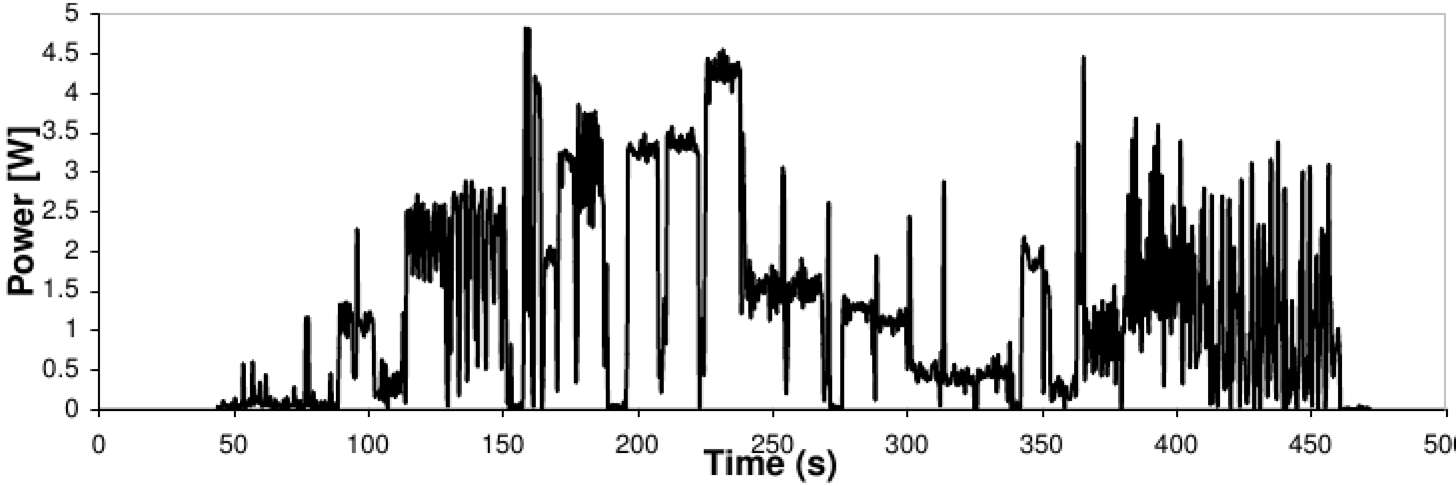
Representative Vectors



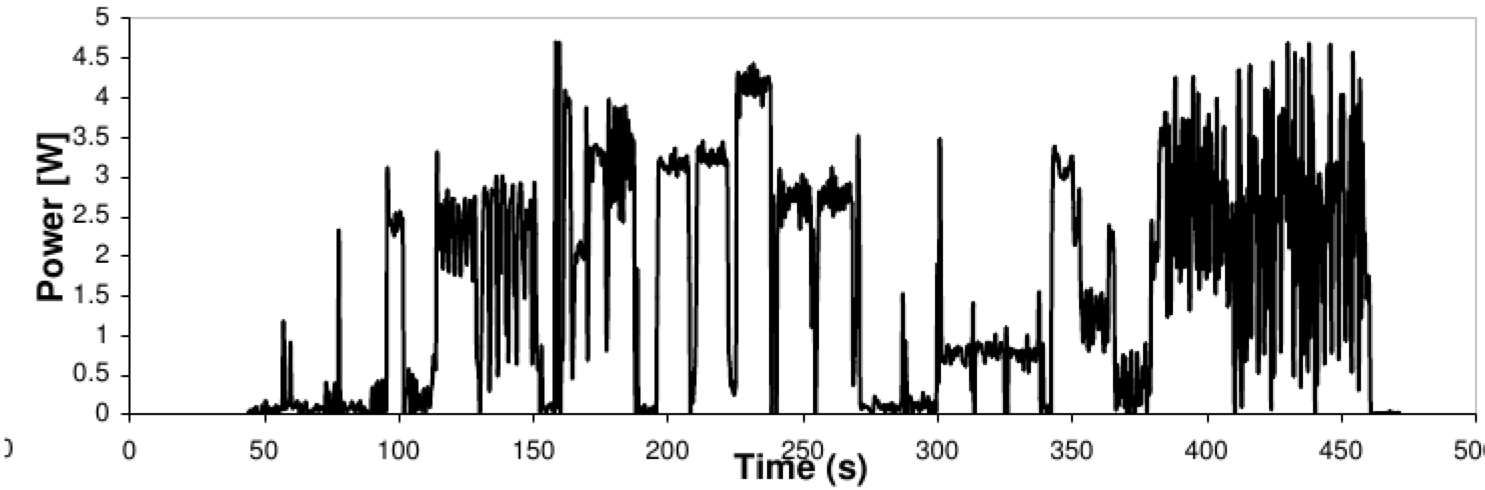


Bus Control
ITLB & Fetch
MEM control
FP Exec
Inst Dec
Ucode ROM
Inst Queue1
RETIRE
L2 Cache
L1 cache
Data TLB
INT Regfile
Trace Cache
Allocation
Inst Queue2
TOTAL_MODELED_POWER
2nd Level BPU
MOB
INT Exec
FP Regfile
1st Level BPU
Rename
Schedule
Vector Components
0
10
20
30
40
50
60
44
66
88
110
132
154
176
198
220
242
264
286
308
330
352
374
396
418
440
462

Error Analysis



(a) Error for representative vectors



(b) Error for selected execution points

Conclusion

- Defined combined similarity metric
- Found only considering total power conceals power phase information
- Grouping of vectors based on thresholding
- Generation of signatures based on representative vectors

Discussion

- Only show errors based on simulation :(
- Nice method to reduce benchmarking time :)