Characterizing the Unique and Diverse Behaviors in Existing and Emerging General-Purpose and Domain-Specific Benchmark Suites

Kenneth Hoste and Lieven Eeckhout
Ghent University, Belgium

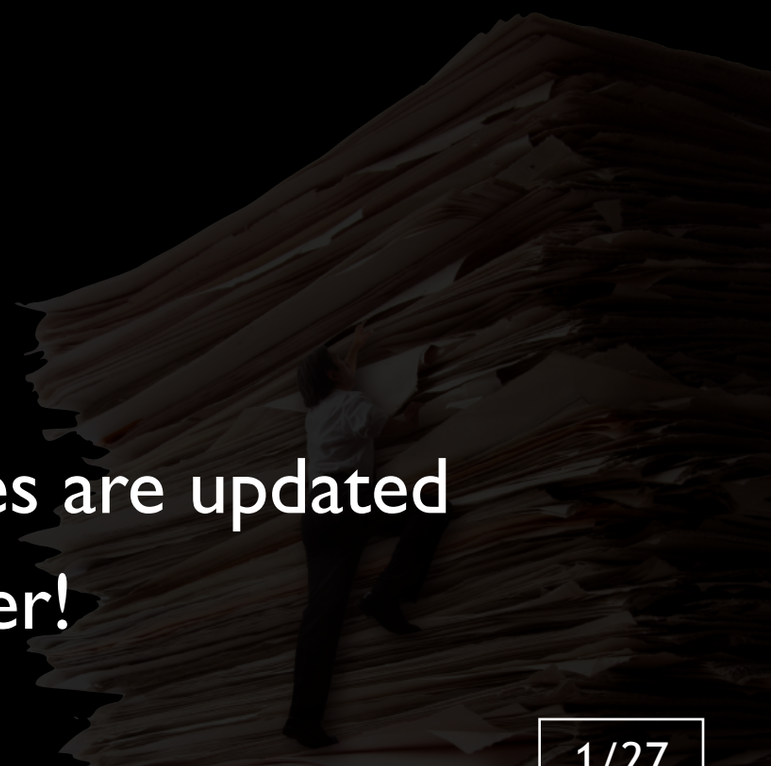
ISPASS-2008, Austin (TX)
April 22th, 2008

UNIVERSITEIT GENT

Benchmarking as common practice

Characterizing the Unique and Diverse Behaviors in Existing and Emerging General-Purpose and Domain-Specific **Benchmark Suites**

- ✦ invaluable to computer architects and researchers
  - ➡ designing future computer systems
  - ➡ evaluating innovations
- ✦ *but :*
  - ➡ ever evolving: new suites emerge, existing ones are updated
  - ➡ huge pressure: we can't keep simulating forever!



Is SPEC CPU enough?

Characterizing the Unique and Diverse Behaviors in Existing and Emerging **General-Purpose and Domain-Specific** Benchmark Suites

general-purpose

- SPEC CPU: compute-intensive workloads
- MiBench (MiDataSets): embedded workloads
- EEMBC: embedded workloads

domain-specific

- BioPerf (bioinformatics), BioMetricsWorkload (biometrics), MediaBenchII (multimedia), MineBench (data mining), PhysicsBench (game physics), ...

Is SPEC CPU enough?

Characterizing the Unique and Diverse Behaviors in Existing and Emerging **General-Purpose and Domain-Specific** Benchmark Suites

general-purpose

- SPEC ute-intensive workloads
- MiBe dded workloads
- EEMBC: emb

How different are domain-specific workloads from general-purpose workloads like SPEC CPU?

domain-specific

- BioPerf (bioinformatics), BioMetrics v ometrics), MediaBenchII (multimedia), MineBench (data mining), PhysicsBench (game physics), ...

New domains, new benchmarks

Characterizing the Unique and Diverse Behaviors in **Existing and Emerging** General-Purpose and Domain-Specific Benchmark Suites

existing (and evolving)

- SPEC CPU: CPU92, CPU95, CPU2000, CPU2006
- MiBench (2001), MiDataSets for MiBench (2007)
- MediaBench (1997), MediaBenchII (2005 - ...)
- EEMBC (1997 - now)

emerging

- BioPerf (2005), BioMetricsWorkoad (2005), MineBench (2005), PhysicsBench (2007)

New domains, new benchmarks

Characterizing the Unique and Diverse Behaviors in **Existing and Emerging** General-Purpose and Domain-Specific Benchmark Suites

existing (and evolving)

- ➡ SPEC ..., CPU95, CPU2000, CPU2006
- ➡ MiBe... for MiBench (2007)
- ➡ MediaBench (...)
- ➡ EEMBC (1997 - now)

Should we add emerging benchmark suites to our current setup to ensure global coverage?

emerging

- ➡ BioPerf (2005), BioMetricsWorkload (...), MineBench (2005), PhysicsBench (2007)

Characterizing dynamic behavior

Characterizing the Unique and Diverse **Behaviors** in Existing and Emerging General-Purpose and Domain-Specific Benchmark Suites

most papers use HPC or simulation metrics

- ➡ IPC, cache miss rates, instruction mix, ...
- ➡ problem: platform-dependent, and so are the conclusions

our approach:

microarchitecture-independent characteristics

- ➡ independent of particular cache configuration, branch predictor, ...
- ➡ captures inherent *phase-level* program behavior

Microarchitecture-Independent Characteristics

overview:

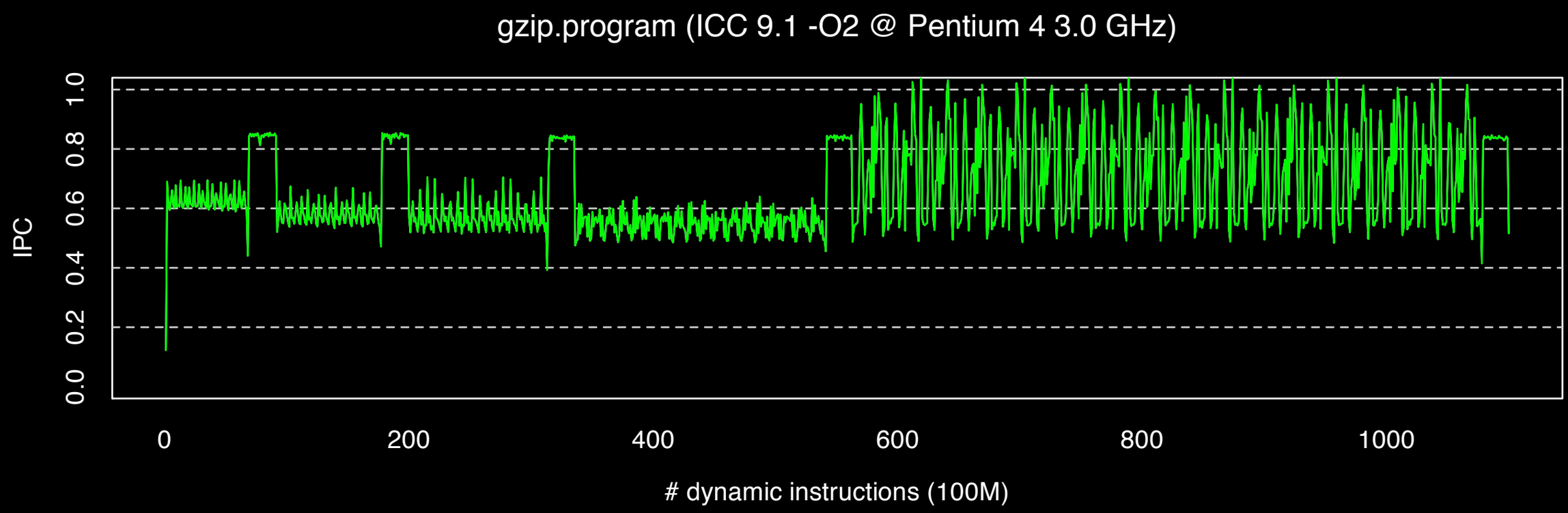
- instruction mix
- amount of inherent ILP for an idealized processor
- register dependency distances, register reuse and # register operands
- memory footprint in # pages and # blocks
- data stream strides (spatial locality)
- branch predictability using theoretical PPM-model

➡ results in *69 characteristics* in total

➡ more info and relevant papers at

http://www.elis.ugent.be/~kehoste/MICA

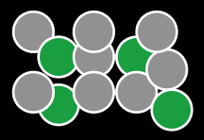
Time changes everything ...



program characteristics are measured per interval of dynamic instructions

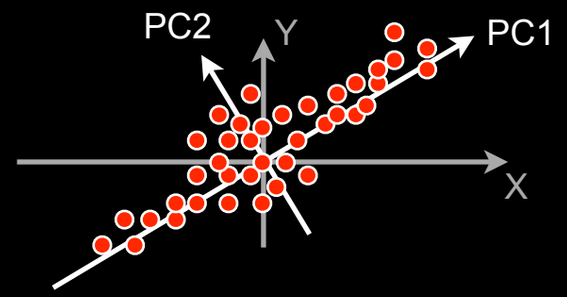
- captures time-variant program behavior ...
- ... but results in a lot of data to cope with
- getting insight is no easy task
- requires post-processing to capture trends and reason about them

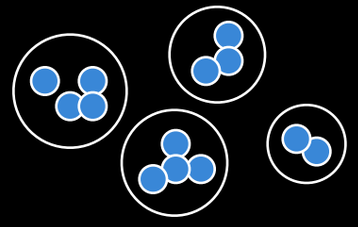
In Short



(1) interval sampling

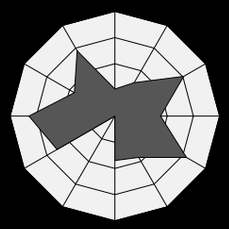
(2) Principal Components Analysis (PCA)





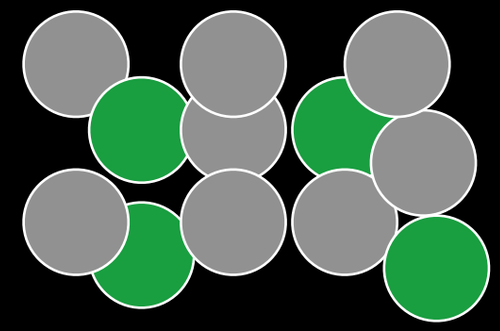
(3) cluster analysis

(4) identify key program characteristics



(5) visualize prominent behaviors using kiviat plots

Sample and conquer

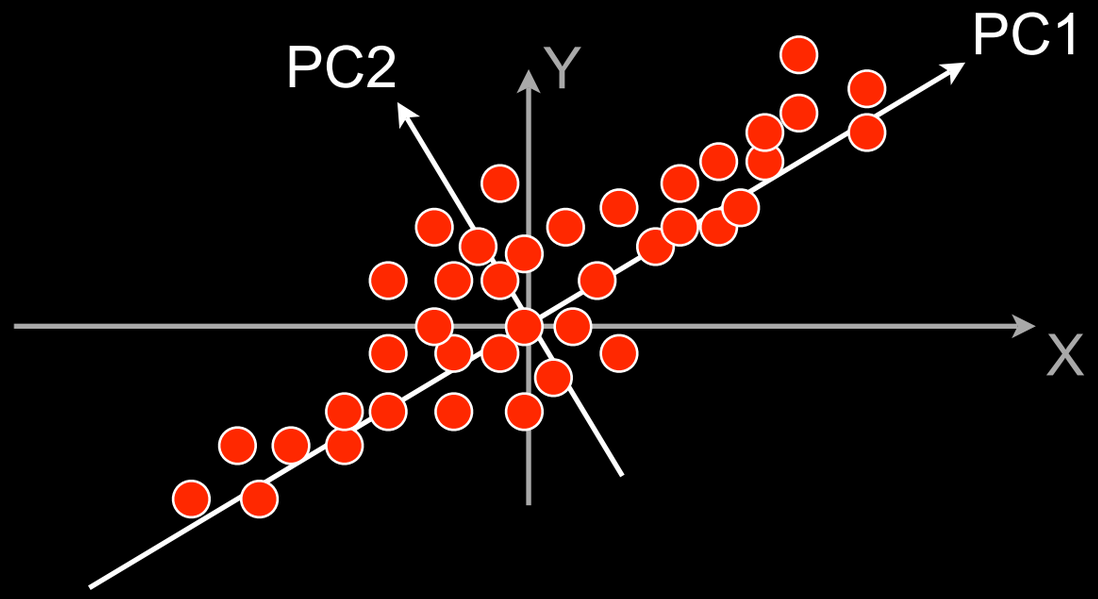


we randomly select a fixed number of intervals per benchmark

- ➡ the total number of intervals is limited
  - ➡ important for subsequent analysis steps
  - ➡ sampled data set still captures overall trends
- ➡ each benchmark gets the same weight
  - ➡ no matter how long it takes to execute, or how many inputs it has
  - ➡ other weighting options possible: per suite, per input, ...
- ➡ in this study: 1,000 intervals per benchmark

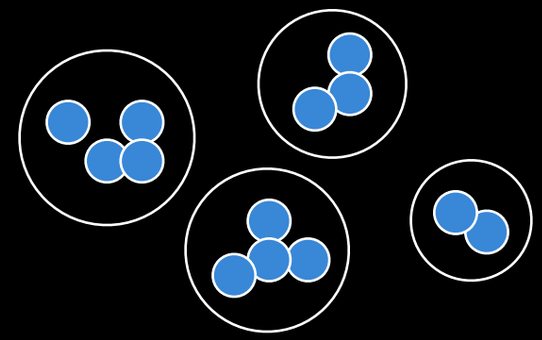
Untangling the characteristics

PCA extracts the underlying trends in the data

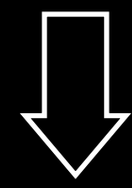


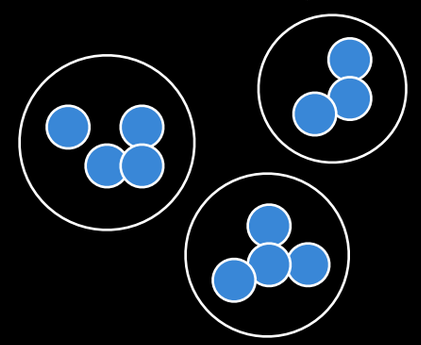
- ➡ resulting dimensions are uncorrelated
- ➡ allows for reducing dimensionality while controlling the amount of information that is lost
- ➡ fast, thus suitable for large data sets

Getting organized



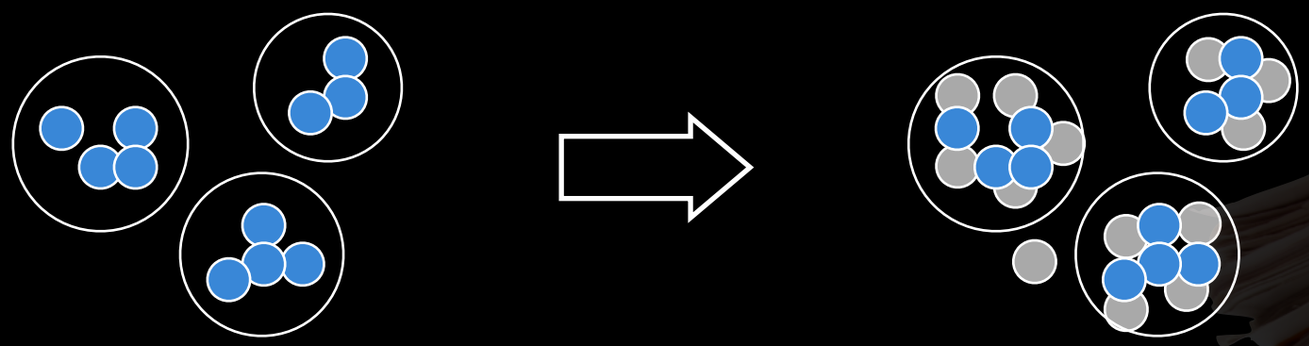
k-means clustering groups intervals into k groups based on similarity





➡ retain largest *n* out of *k* clusters
⇒ trade off in-cluster variability vs coverage

➡ map clusters obtained for sampled data onto full data set
⇒ evaluate using all data, not just sampled data

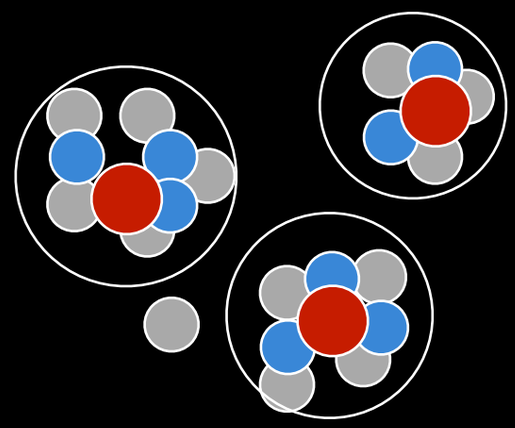


➡ clustering on full data set too time consuming
⇒ approximation through sampling makes it feasible

Finding the key

determining the key program characteristics

- ➡ obtain a limited number of characteristics which correlate well with the full set of characteristics
- ➡ approximate distances between intervals
- ➡ based on characteristics for the cluster representatives

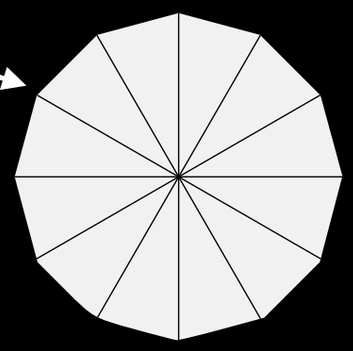


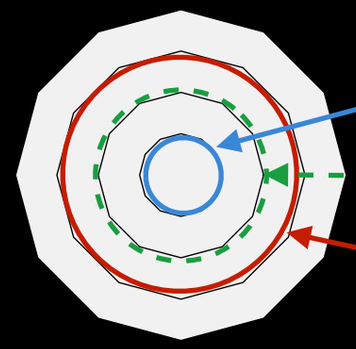
- ➡ details: see our IISWC-2006 paper
  "Comparing Benchmarks Using Key Microarchitecture-Independent Characteristics"

Visualize to analyze

each of the cluster representatives is visualized

each axis corresponds
to one key characteristic





- using kiviat plots (radar plots, ...)
- in terms of the key program characteristics

  ⇒ easy to interpret

  ⇒ easy to compare prominent behaviors

- grouped by type: suite-specific, benchmark-specific, ...

Putting it to the test

5 benchmark suites:

SPEC CPU2000 (ref), SPEC CPU2006 (ref), BioPerf (medium), BioMetricsWorkload (s100), MediaBenchII

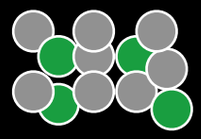
77 benchmarks

69 program characteristics per interval

1,103,953 100M-instruction intervals

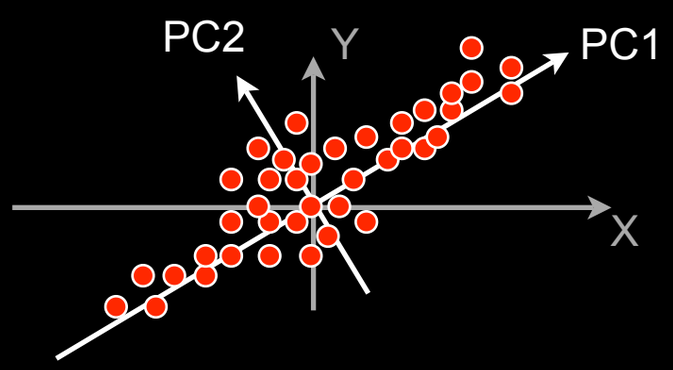
⇒ over 800M of data to cope with

Step by step (1)

(1) interval sampling 

randomly picking 1,000 intervals per benchmark (cross-input)

⇒ 77,000 intervals to run subsequent steps on

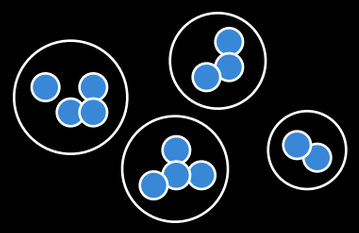


(2) Principal Components Analysis

retaining all PCs with std. dev. > 1
⇒ 13 PCs explaining 85.4% of total variance

significant reduction of dimensionality

Step by step (2)



(3) cluster analysis

cluster into 300 clusters, only retaining 100 largest

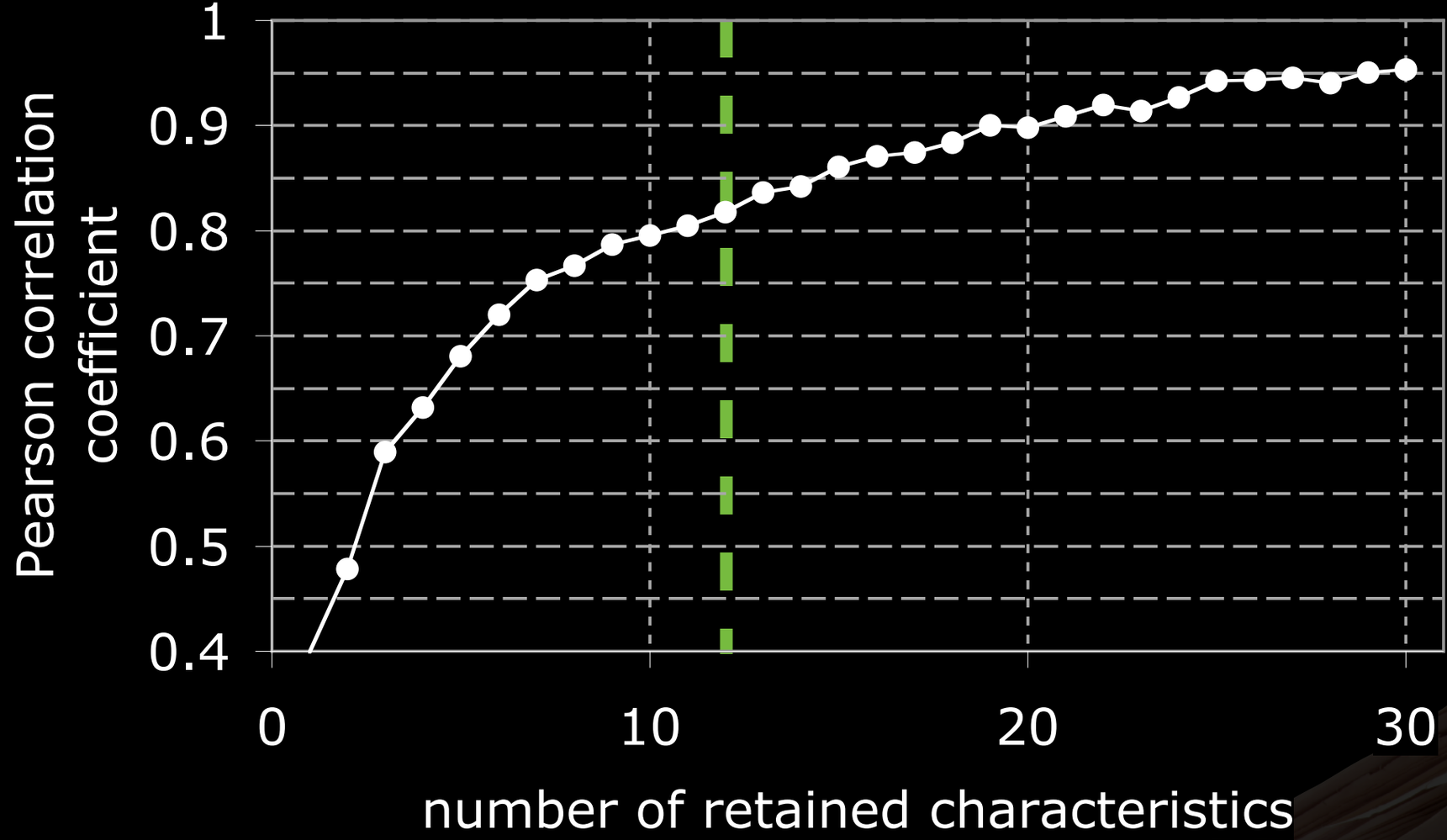
⇒ covers 87.8% of total workload space

with significant reduction of in-cluster variability

(4) identifying key program characteristics

run genetic algorithm on 100 cluster representatives, using the original 69 characteristics

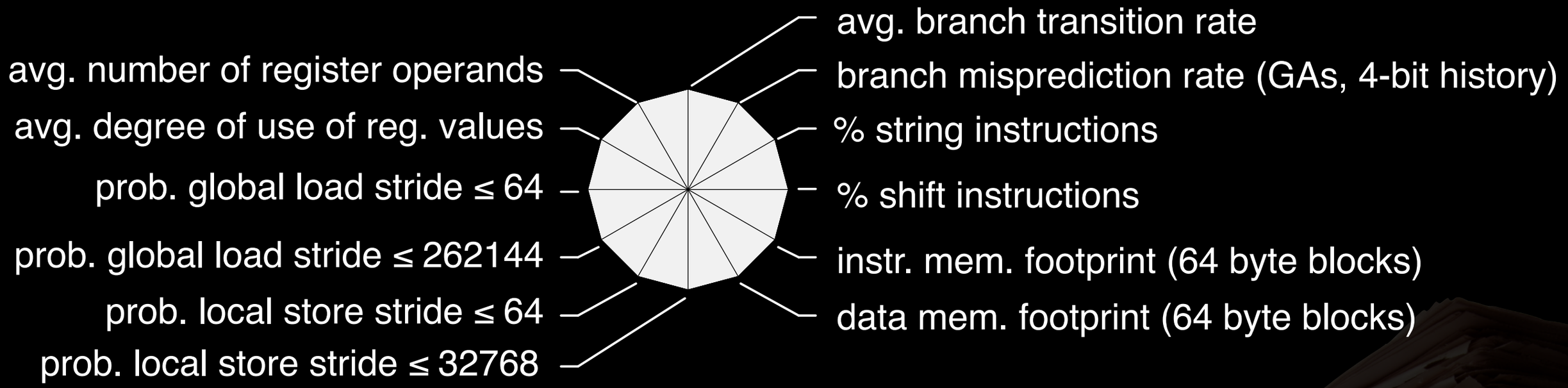
Step by step (3)

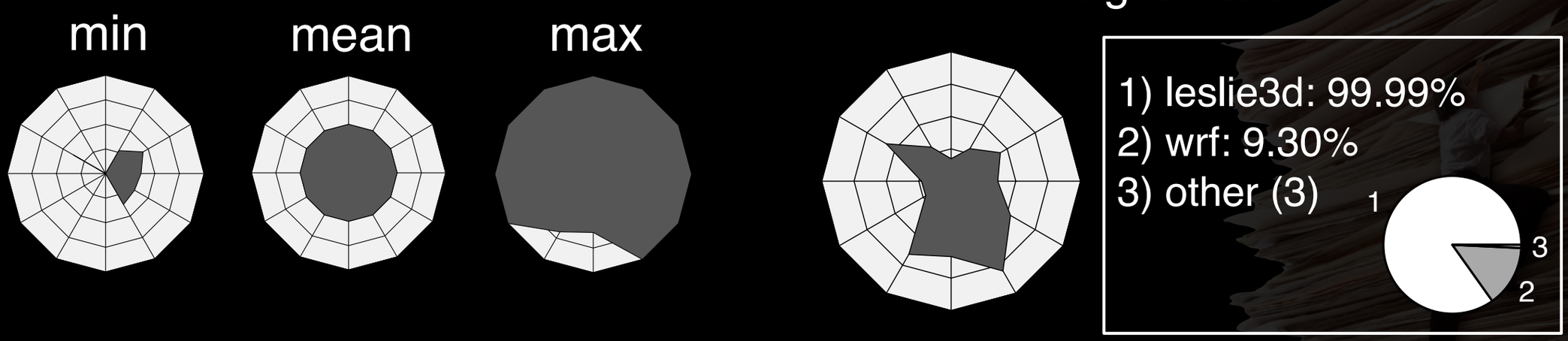


just 12 characteristics correlate with all 69 characteristics with a correlation coefficient of 0.82

Step by step (4)

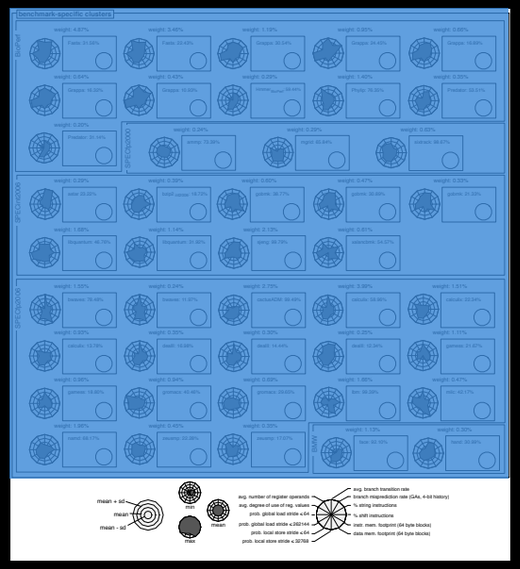
(5) visualize prominent behaviors using kiviat plots





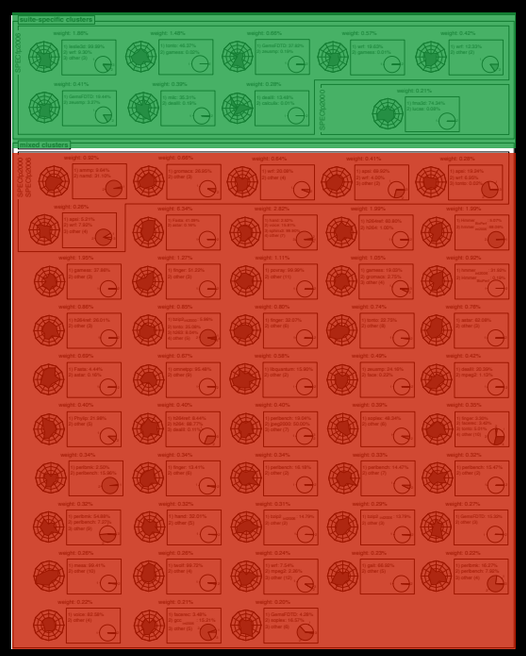
UNIVERSITEIT GENT

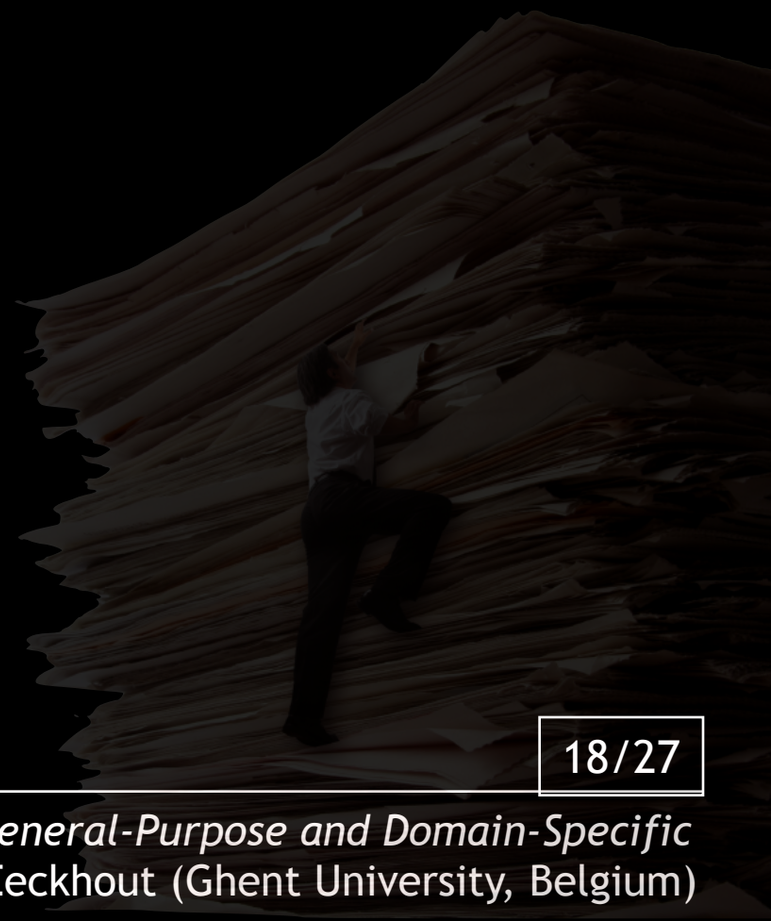
Interpretation made easy



grouping of kiviat plots

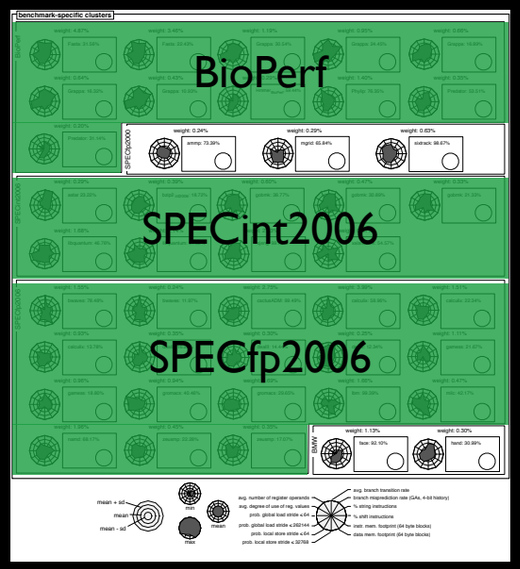
- ➡ benchmark-specific (■)
- ➡ suite-specific (■)
- ➡ mixed (■)





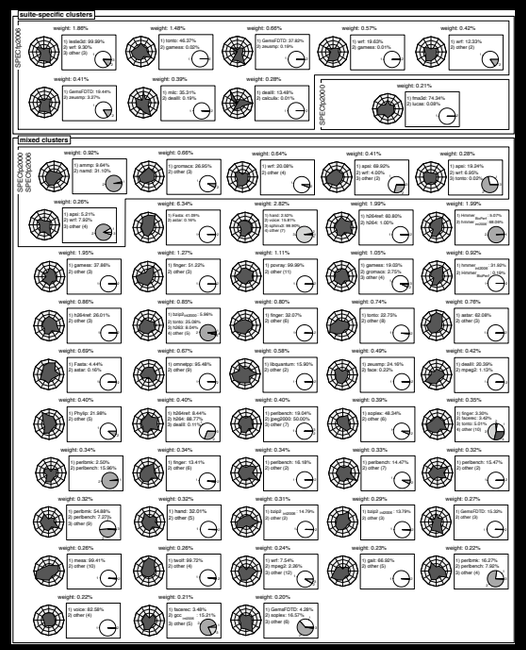
UNIVERSITEIT GENT

Interpretation made easy

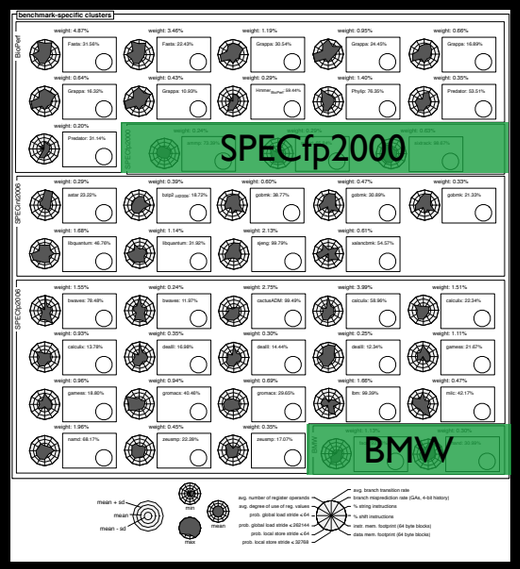


benchmark-specific prominent behaviors

➡ SPECint2006, SPECfp2006, BioPerf

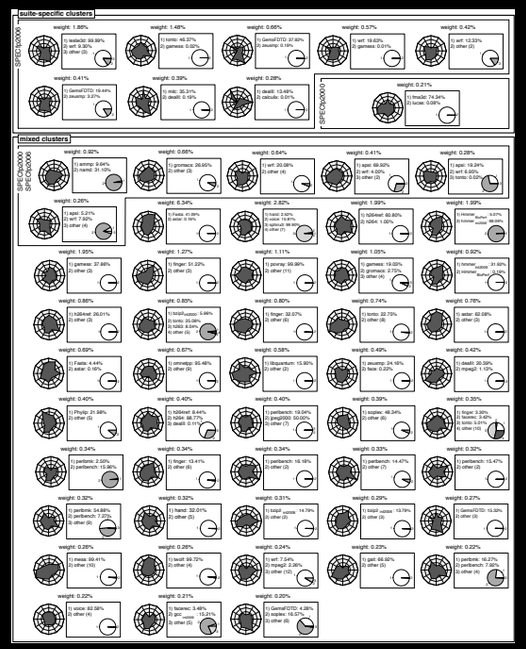


Interpretation made easy



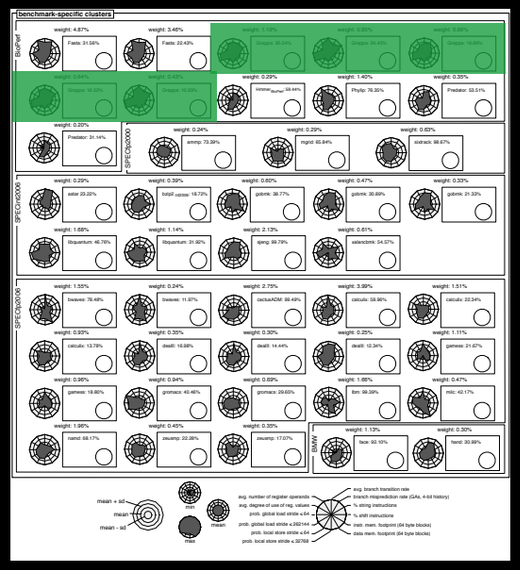
benchmark-specific prominent behaviors

- ➡ SPECint2006, SPECfp2006, BioPerf
- ➡ SPECfp2000, BioMetricsWorkload (less)



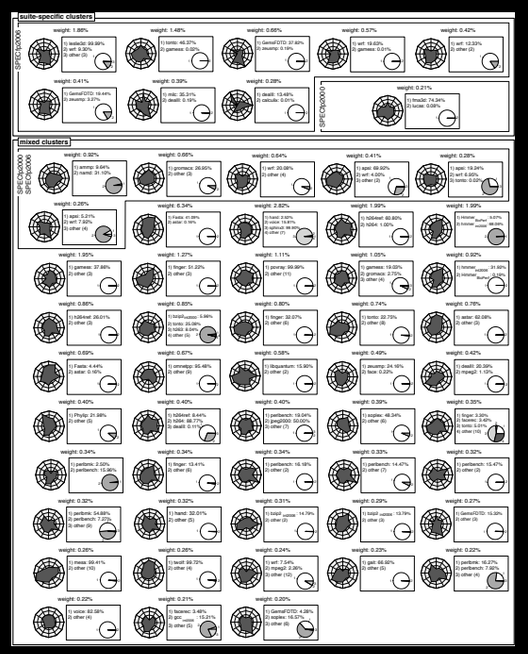
UNIVERSITEIT GENT

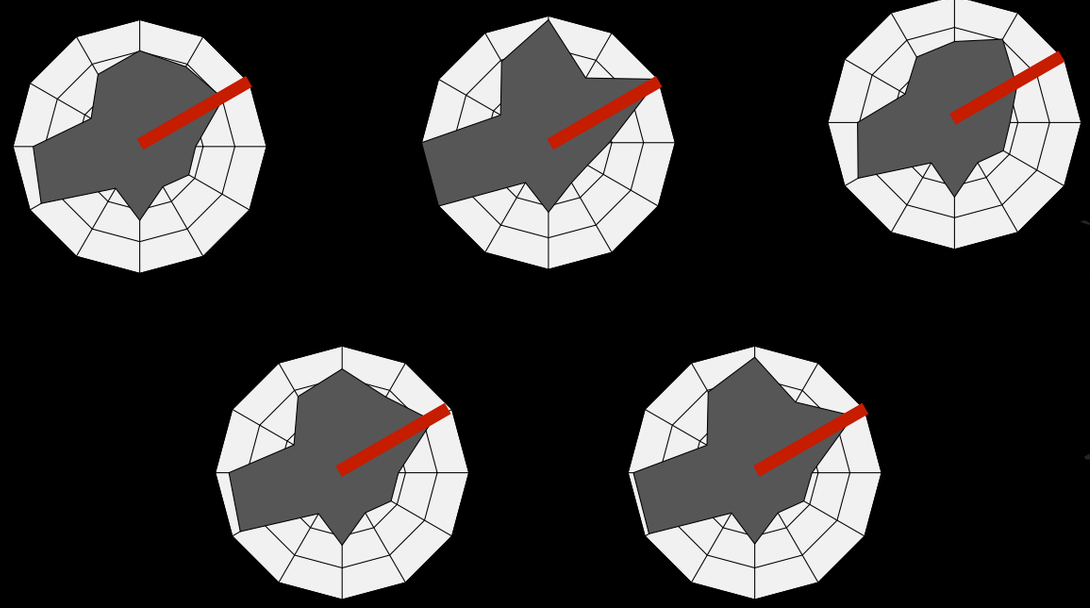
Interpretation made easy



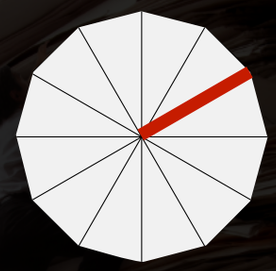
benchmark-specific prominent behaviors

- ➡ SPECint2006, SPECfp2006, BioPerf
- ➡ SPECfp2000, BioMetricsWorkload (less)
- ➡ Grappa differs from the others



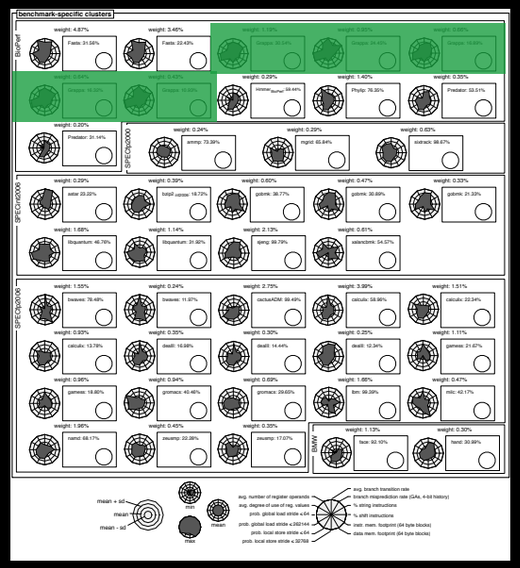


string instr.



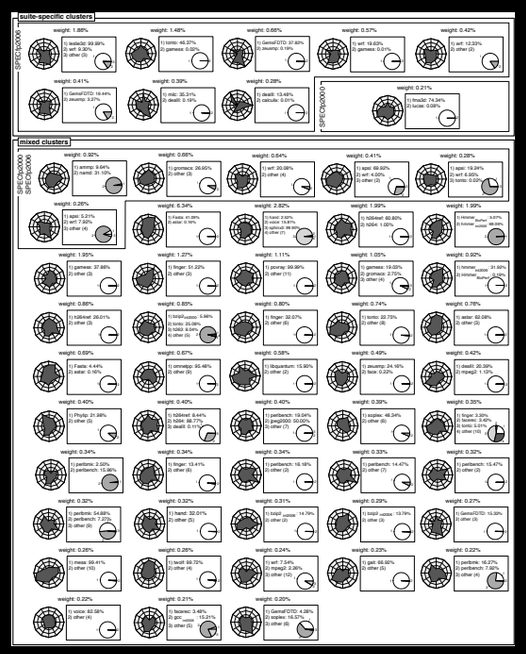
UNIVERSITEIT GENT

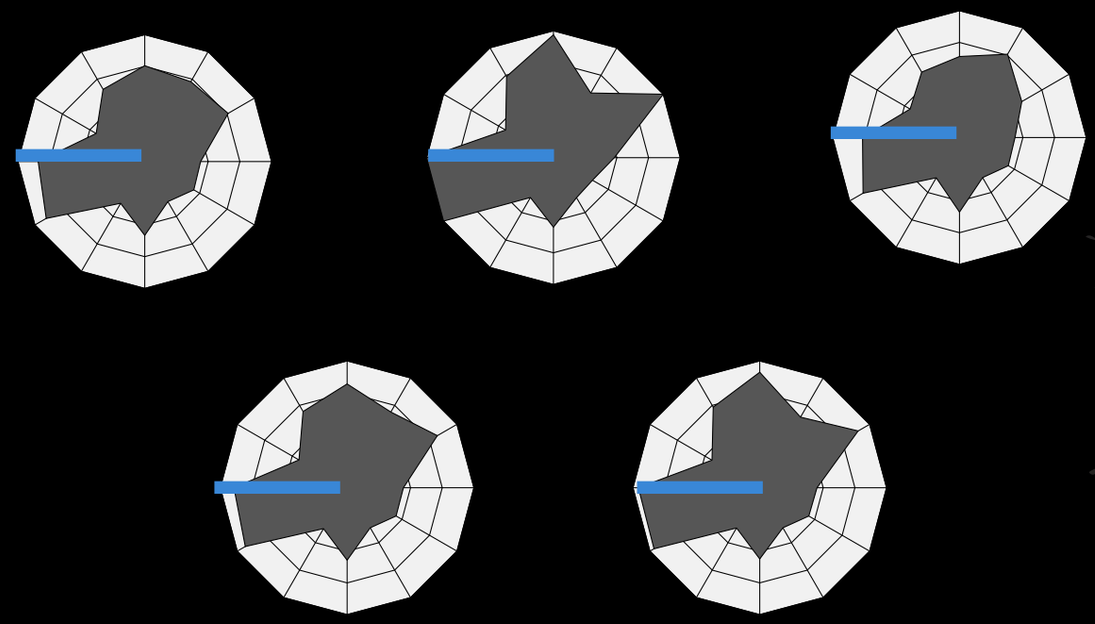
Interpretation made easy



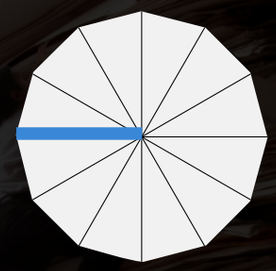
benchmark-specific prominent behaviors

- ➡ SPECint2006, SPECfp2006, BioPerf
- ➡ SPECfp2000, BioMetricsWorkload (less)
- ➡ Grappa differs from the others





global load
stride ≤ 64

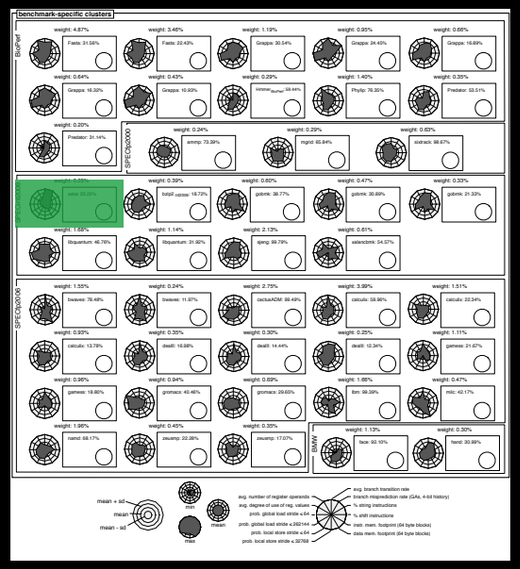


UNIVERSITEIT GENT

Characterizing the Unique and Diverse Behaviors in Existing and Emerging General-Purpose and Domain-Specific Benchmark Suites

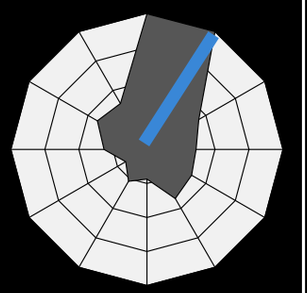
Kenneth Hoste and Lieven Eeckhout (Ghent University, Belgium)

Interpretation made easy



(suite-specific)

weight: 0.29%

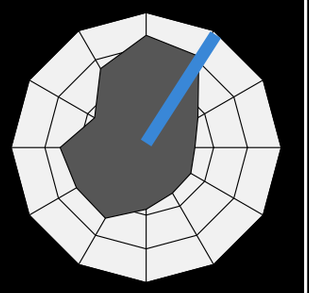


astar 23.22%

BAD

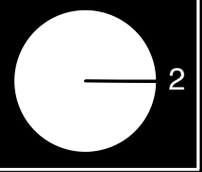
(mixed)

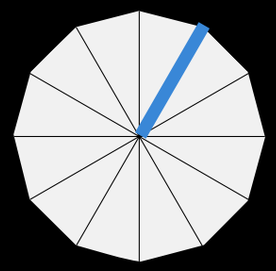
weight: 0.76%



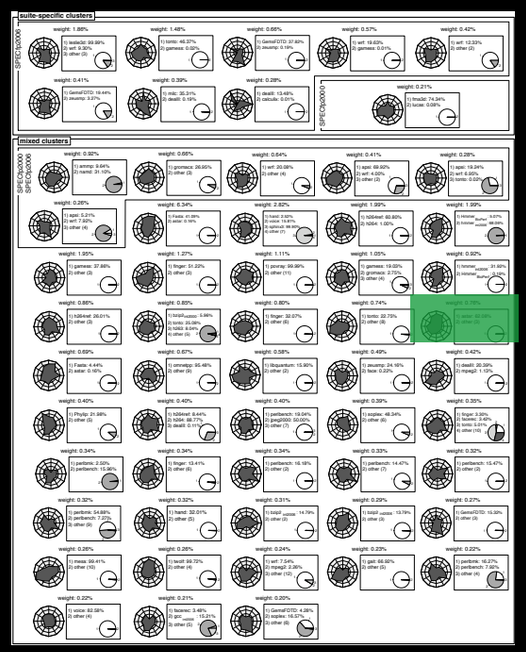
1) astar: 62.08%
2) other (3)

GOOD





branch predictability

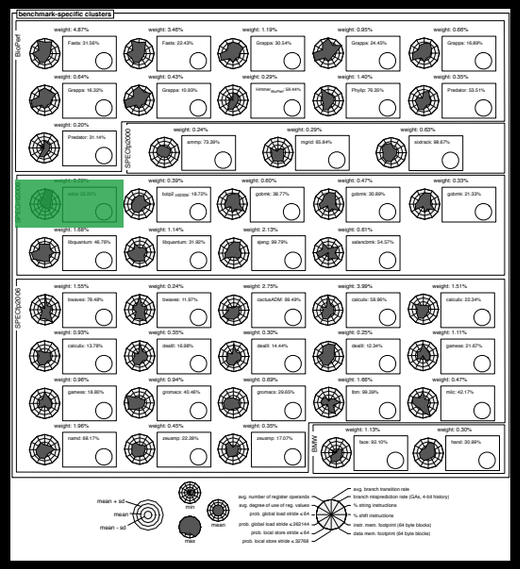


quick insight into dynamic behavior

➡ astar shows two clearly different types of phases

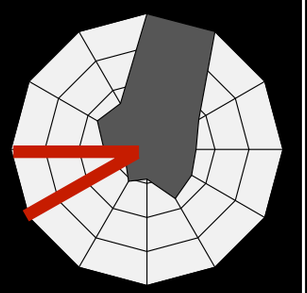
UNIVERSITEIT GENT

Interpretation made easy



(suite-specific)

weight: 0.29%

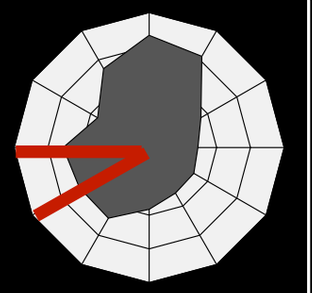


astar 23.22%

BAD

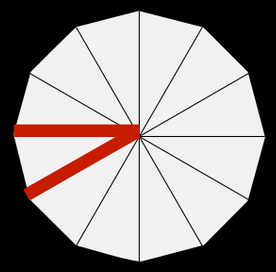
(mixed)

weight: 0.76%

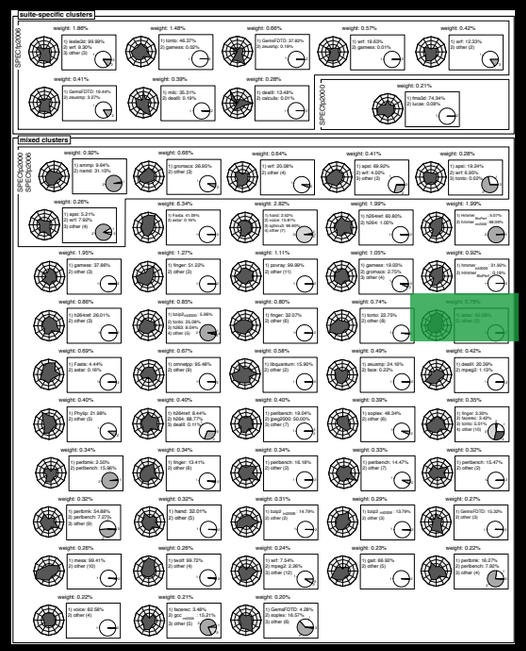


1) astar: 62.08%
2) other (3)

GOOD 1 2



data locality (global load strides)

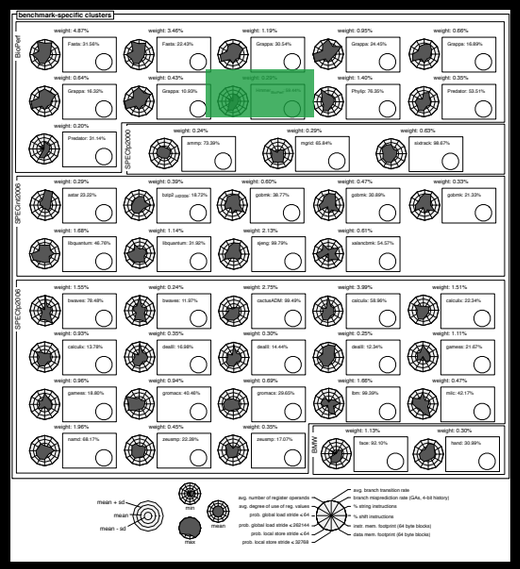


quick insight into dynamic behavior

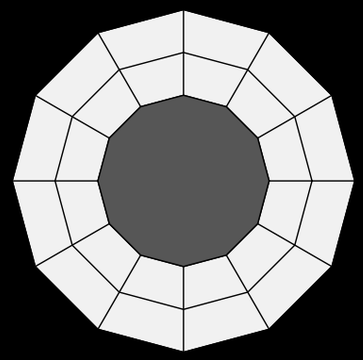
➡ astar shows two clearly different types of phases

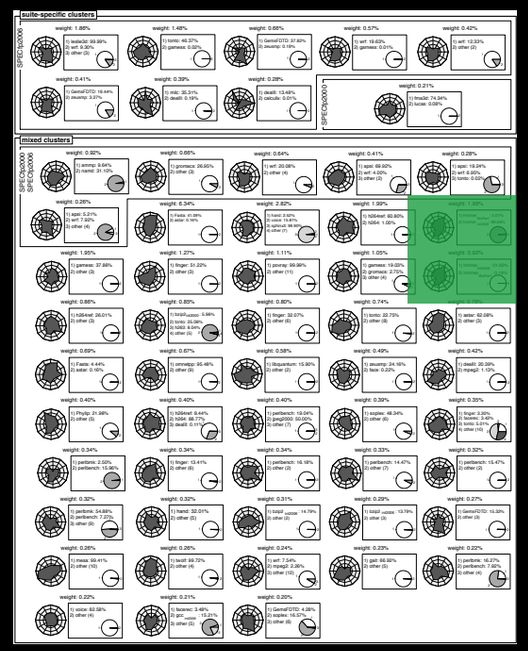
UNIVERSITEIT GENT

Interpretation made easy



mean





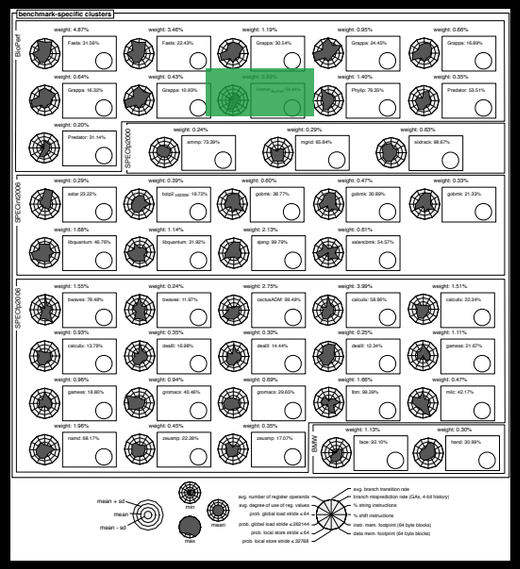
quick insight into dynamic behavior

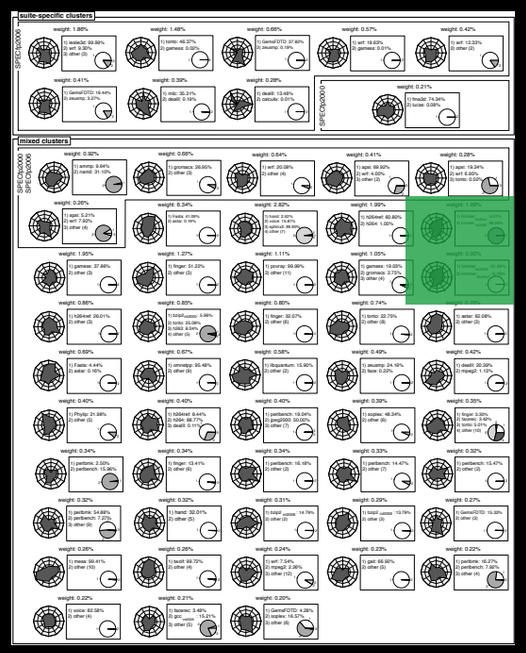
➡ astar shows two clearly different types of phases

mixed behaviors are more average

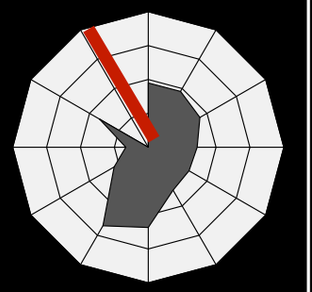
UNIVERSITEIT GENT

Interpretation made easy

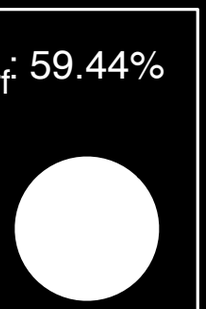




weight: 0.29%

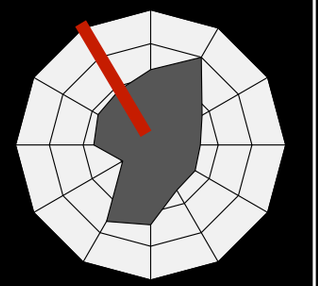


Hmmer $_{\text{BioPerf}}$: 59.44%

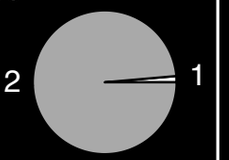


register operands

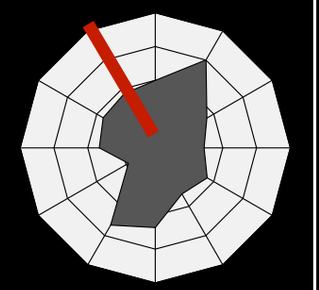
weight: 1.99%



1) Hmmer $_{\text{BioPerf}}$: 5.07%
2) hmmer $_{\text{int2006}}$: 68.06%



weight: 0.92%



1) hmmer $_{\text{int2006}}$: 31.92%
2) Hmmer $_{\text{BioPerf}}$: 0.19%

quick insight into dynamic behavior

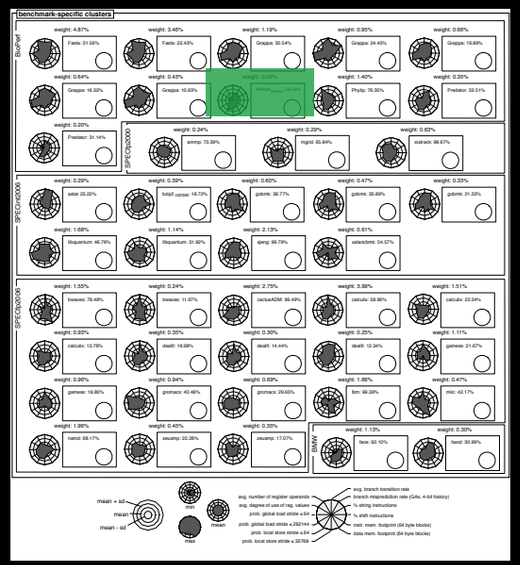
➡ astar shows two clearly different types of phases

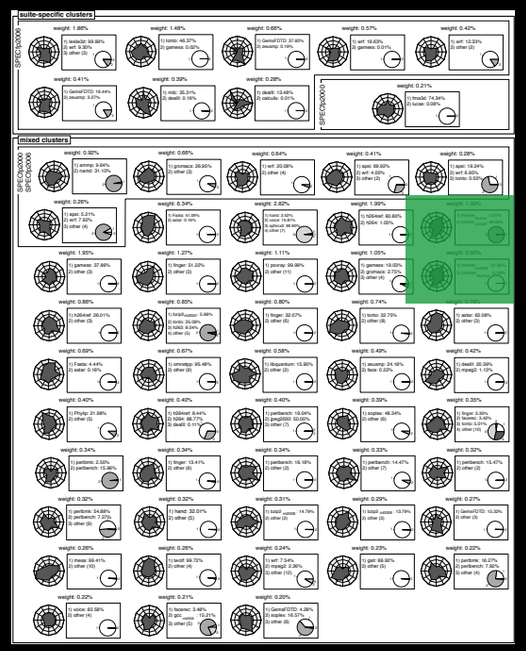
mixed behaviors are more average

➡ hmmer behavior different across suites

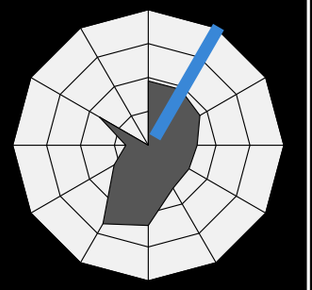
UNIVERSITEIT GENT

Interpretation made easy



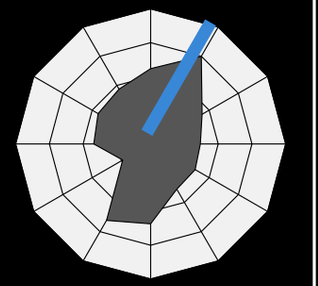


weight: 0.29%

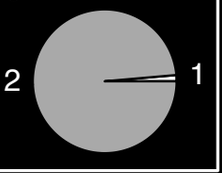


Hmmer$_{\text{BioPerf}}$: 59.44%

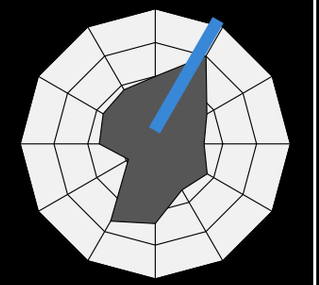
weight: 1.99%



1) Hmmer$_{\text{BioPerf}}$: 5.07%
2) hmmer$_{\text{int2006}}$: 68.06%



weight: 0.92%



1) hmmer$_{\text{int2006}}$: 31.92%
2) Hmmer$_{\text{BioPerf}}$: 0.19%

branch predictability

quick insight into dynamic behavior

➡ astar shows two clearly different types of phases

mixed behaviors are more average

➡ hmmer behavior different across suites

UNIVERSITEIT GENT

On the side: coverage, diversity and uniqueness

Characterizing the **Unique and Diverse** Behaviors in Existing and Emerging General-Purpose and Domain-Specific Benchmark Suites

coverage:

how many clusters contain intervals of suite S ?

diversity:

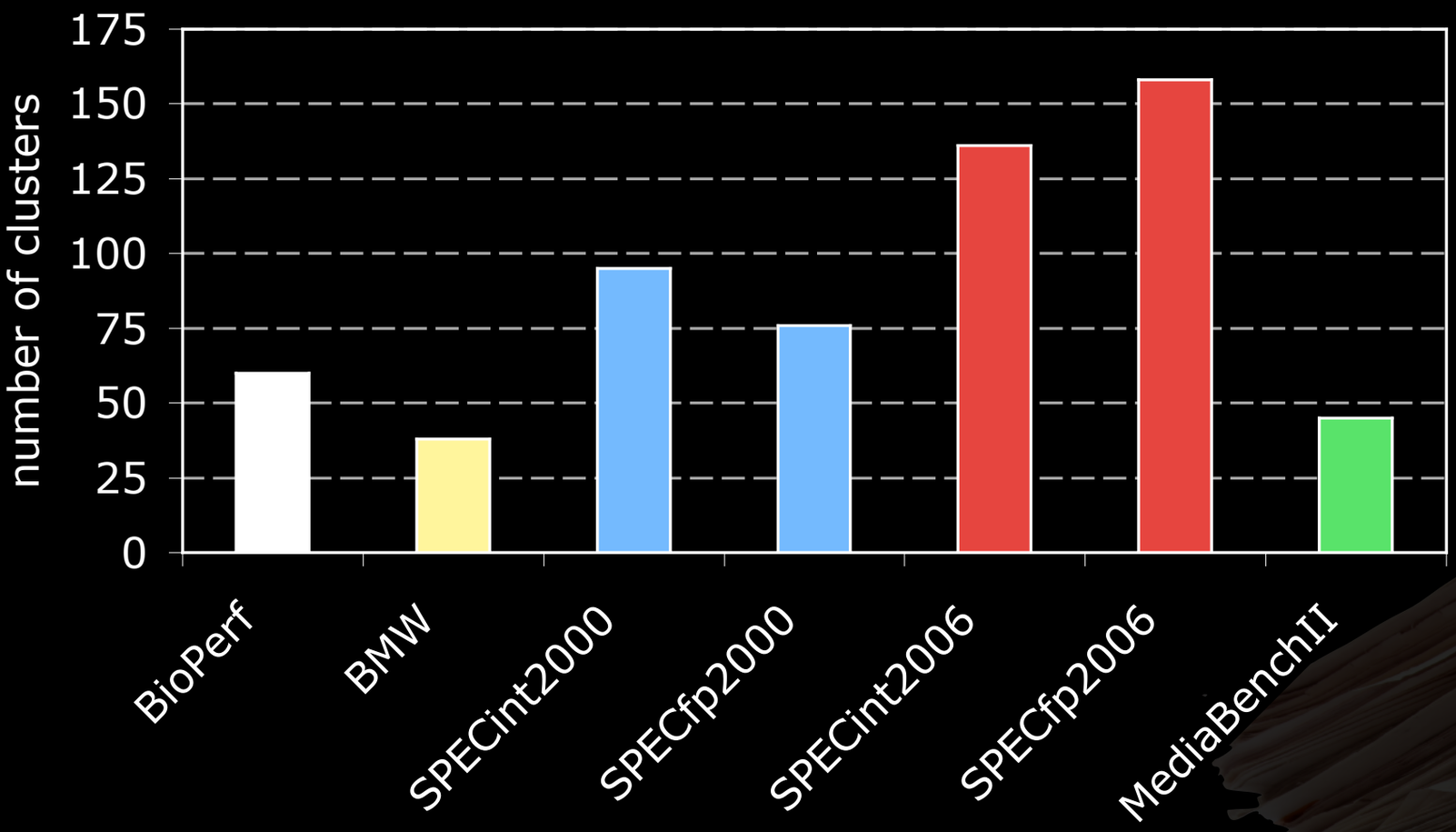
how many clusters do we need to capture most of suite S ?

unique behavior:

how many clusters contain *only* intervals of suite S ?

Coverage

How many clusters contain intervals of suite S ?

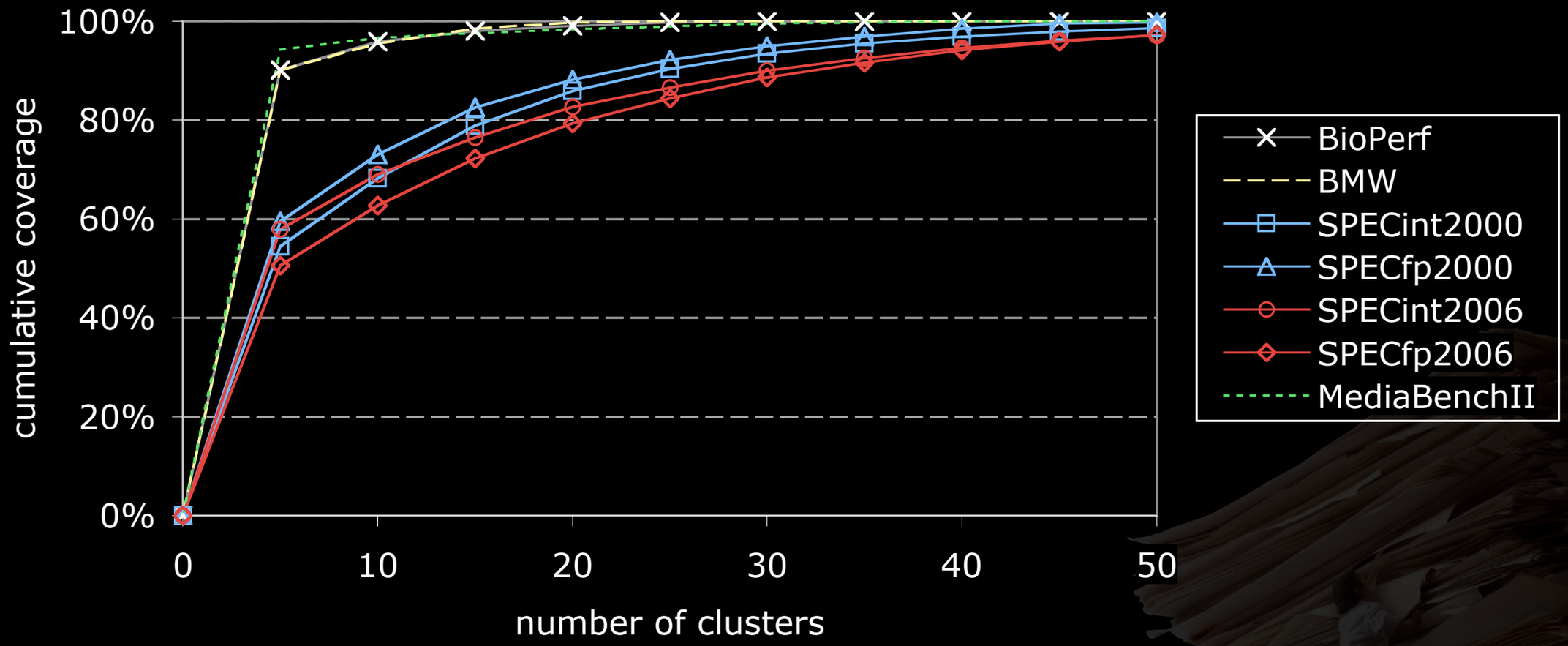


domain-specific suites cover a much narrower part of the workload space

UNIVERSITEIT GENT

Diversity

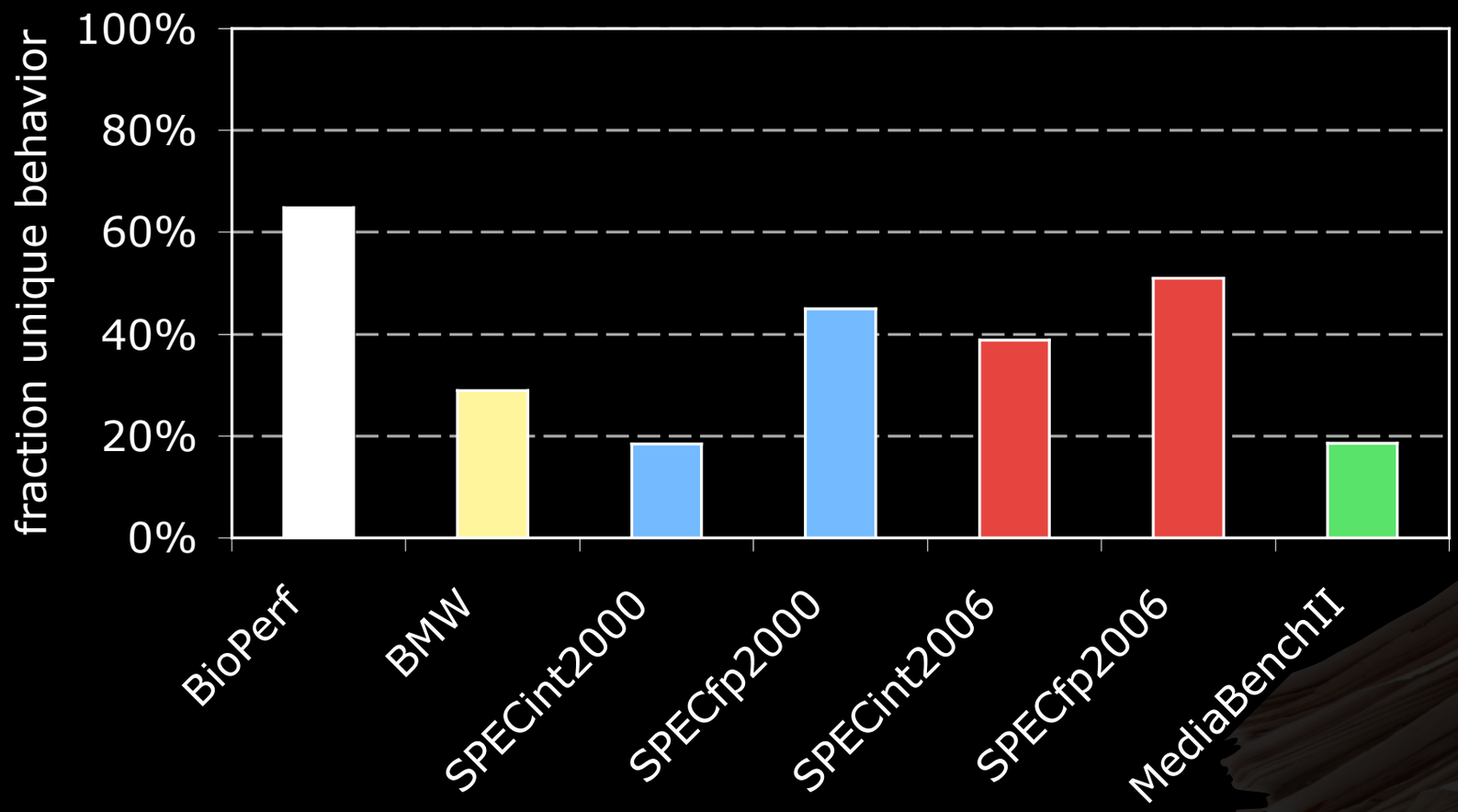
How many clusters do we need to capture most of suite S ?



domain-specific suites also show less diverse behavior within the suite itself

Unique behavior

How many clusters contain *only* intervals of suite S ?



some domain-specific suites may show a significant amount of behavior not in any other suite

Lessons learned

- SPEC CPU2006 shows broader coverage of workload space compared to CPU2000
- CPU2006 is only slightly more diverse than its predecessor

  ⇒ *slightly* larger number of samples should be enough
- BioPerf shows a significant amount of unique behavior

  ⇒ good suite to also take into account in analysis
- SPEC CPU2000 is still important to take into account, next to CPU2006

  ⇒ a lot of its behavior is not represented in CPU2006

Conclusions

microarchitecture-independent phase-level analysis made feasible

- ➡ from over 1M instruction intervals to just 100 easily interpretable visual representations of most prominent phase behaviors
- ➡ captures important patterns for benchmark (suite) comparison
- ➡ various interesting insights in the blink of an eye

assessing unique and diverse behavior

- ➡ quantifies intuitions and reveals importance of emerging suites
- ➡ leads to guidelines for selecting benchmark suites

Characterizing the Unique and Diverse Behaviors in Existing and Emerging General-Purpose and Domain-Specific Benchmark Suites

Kenneth Hoste and Lieven Eeckhout
Ghent University, Belgium

ISPASS-2008, Austin (TX)
April 22th, 2008

UNIVERSITEIT GENT