Identifying the best performing hardware platform based on inherent program similarity

Kenneth Hoste, Lieven Eeckhout, Andy Georges and Koen De Bosschere

Ghent University, Belgium

Goal

efficiently identifying the best hardware platform for an application of interest

Problem

native execution/simulation not feasible

- porting problems
- hardware might not be available (yet)
- too time-consuming

current practice: choose the hardware platform with the best average performance

Methodology

- characterize programs independent from microarchitecture
- estimate performance based on inherent program similarity
- rank machines based on estimated performance

47 microarchitecture-independent program characteristics

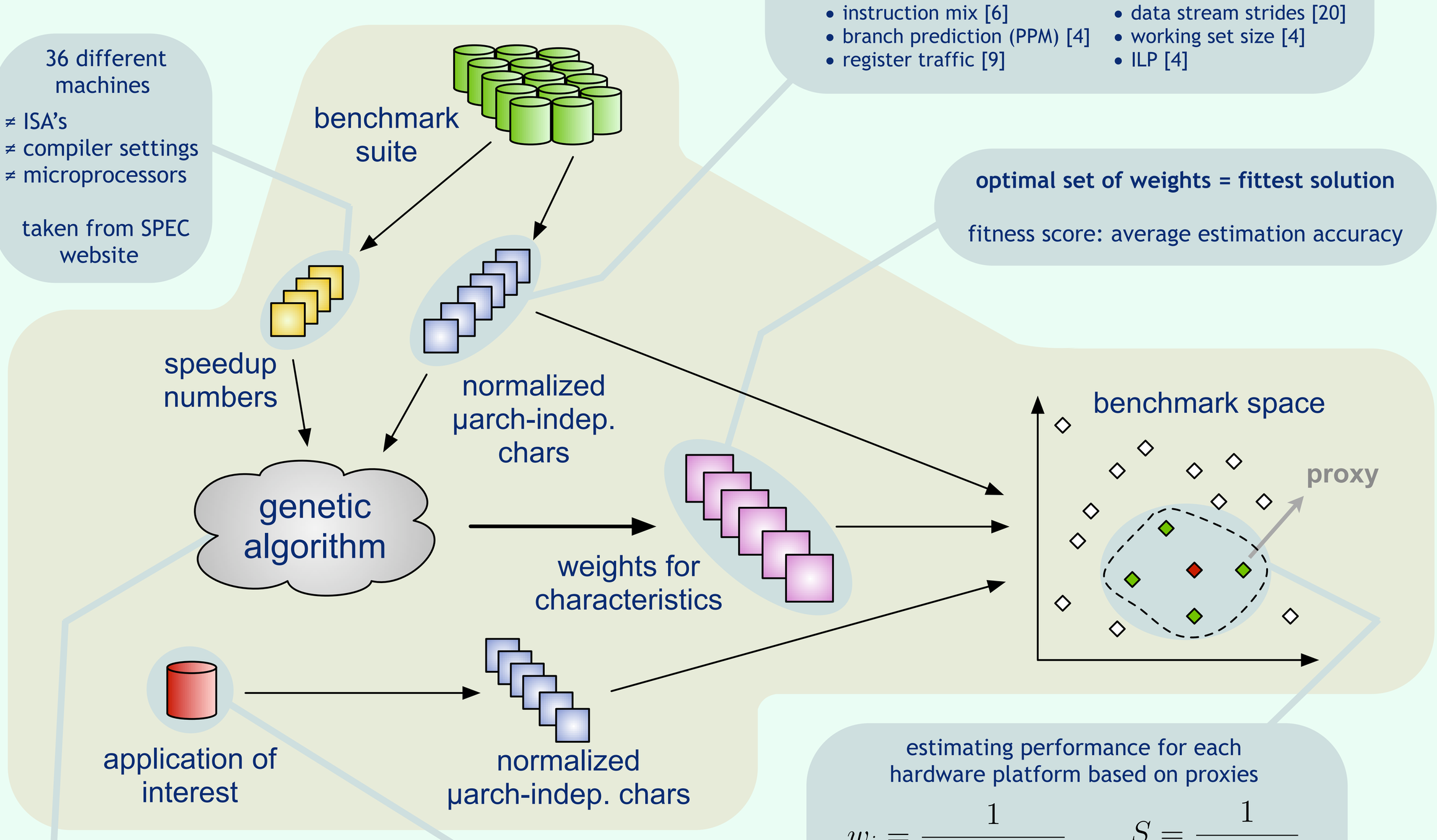
6 categories:

- instruction mix [6]
- branch prediction (PPM) [4]
- register traffic [9]
- data stream strides [20]
- working set size [4]
- ILP [4]

36 different machines

≠ ISA's
≠ compiler settings
≠ microprocessors

taken from SPEC website



optimal set of weights = fittest solution

fitness score: average estimation accuracy

estimating performance for each hardware platform based on proxies

$$w_i = \frac{1}{d_i \cdot \sum_{i=1}^{n} \frac{1}{d_i}} \qquad S = \frac{1}{\sum_{i=1}^{n} \frac{w_i}{S_i}}$$

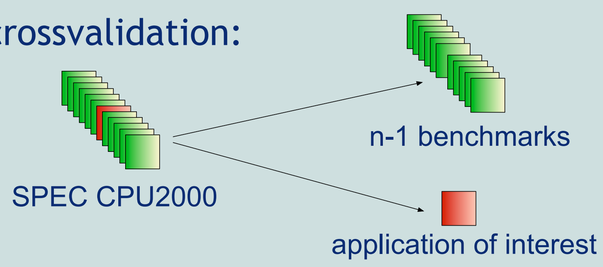
w_i: weight for proxy i; S_i: speedup for proxy i;
d_i: Euclidean distance between app. and proxy i

evolutionary optimization method

building new generations of solutions using *crossover* and *mutation* operators

evaluation using full SPEC CPU2000 benchmark suite

crossvalidation:



one benchmark serves as application of interest, other benchmarks serve as benchmark suite

Evaluation

- ranking machines for each SPEC CPU2000 benchmark based on performance estimations
- evaluating machine rank estimation using Spearman rank coefficient

	average	worst
current practice	0.83	0.64 (*art*)
our approach	0.89	0.79 (*apsi*)

choosing the best machine
(perf. loss compared to real best machine)

- current practice: 20%
- our approach: 13.6%

- *bzip2* is a frequent proxy ⇒ average behavior
- *gcc*, *mcf*, *swim* never appear as proxy ⇒ very specific behavior

[1] K. Hoste, A. Phansalkar, L. Eeckhout, A. Georges, L. K. John and K. De Bosschere. *Performance Prediction based on Inherent Program Similarity*. Accepted for the 15th International Conference on Parallel Architectures and Compilation Techniques (PACT-2006), Sept. 2006.