Statistical Inference for Efficient Microarchitectural and Application Analysis

Benjamin C. Lee

www.deas.harvard.edu/∼bclee
Division of Engineering and Applied Sciences
Harvard University

15 November 2006

Acknowledgements

- David Brooks, Harvard University
- Bronis de Supinski, Lawrence Livermore National Laboratory
- Martin Schulz, Lawrence Livermore National Laboratory

Outline

Outline

Parameter Space Exploration

- **Cycle-Accurate Simulation**
  - Tracks instructions' progress through microprocessor
  - Estimates performance, power, temperature, . . .
- **Execution-Based Profiling**
  - Selectively execute application with varying inputs
  - Estimates performance
- **Exploration Costs**
  - Non-trivial simulation, profiling times
  - Parameter space scales exponentially (m^p)
    - p :: parameter count
    - m :: parameter resolution

Exploration Paradigm

- **Comprehensively understand parameter space**
  - Specify large, high-resolution parameter space
  - Consider all parameters simultaneously
- **Selectively measure modest number of points**
  - Sample points randomly from space for measurement
  - Decouple resolution of space and measurements
- **Efficiently leverage measured data with inference**
  - Reveal trends, trade-offs from sparse sampling
  - Enable predictions for metrics of interest

Model Formulation

- **Notation**
  - n observations ▷ {*measured samples*}
  - Response :: $\vec{y} = y_1, \ldots, y_n$ ▷ {*e.g., performance*}
  - Predictor :: $\vec{x}_i = x_{i,1}, \ldots, x_{i,p}$ ▷ {*e.g., L2 cache*}
  - Regression Coefficients :: $\vec{\beta} = \beta_0, \ldots, \beta_p$
  - Random Error :: $\vec{e} = e_1, \ldots, e_n$ where $e_i \sim N(0, \sigma^2)$
  - Transformations :: $f, \vec{g} = g_1, \ldots, g_p$
- **Model**

$$f(y) = \beta_0 + \sum_{j=1}^{p} \beta_j g_j(x_j) + e$$

Predictor Interaction

- **Modeling Interaction**
  - Suppose effects of predictors x_1, x_2 cannot be separated
  - Construct predictor $x_3 = x_1 x_2$

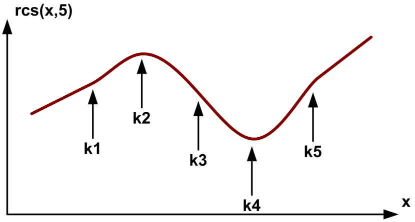
$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2 + e_i$$

- **Example**
  - Let x_1 be pipeline depth, x_2 be L2 cache size
  - Performance impact of pipelining affected by cache size

$$Speedup = \frac{Depth}{1 + Stalls/Inst}$$

Predictor Non-Linearity

- **Restricted Cubic Splines**
  - Divide predictor domain into intervals separated by knots
  - Piecewise cubic polynomials joined at knots
  - Higher order polynomials provide better fits



[2] Stone [SS'86]

Prediction

- **Expected Response**
  - β are known from least squares
  - $x_{i,1}, \ldots, x_{i,p}$ are known for a given query i
  - Expected response is weighted sum of predictor values

$$
\begin{aligned}
E\left[y\right] &= E\left[\beta_0 + \sum_{j=1}^{p} \beta_j x_j\right] + E\left[e\right] \\
&= \beta_0 + \sum_{j=1}^{p} \beta_j x_j
\end{aligned}
$$

Outline

Tools and Benchmarks

- **Simulation Framework**
  - Turandot :: a cycle-accurate trace driven simulator
  - PowerTimer :: power models derived from circuit analyses
  - Baseline simulator models POWER4/POWER5 architecture
- **Benchmarks**
  - SPEC2kCPU :: compute-intensive benchmarks
  - SPECjbb :: Java server benchmark
- **Statistical Framework**
  - R :: software environment for statistical computing
  - Hmisc and Design packages

Predictors :: Microarchitecture

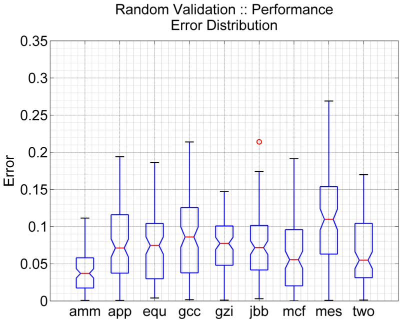
	Set	Parameters	Measure	Range	$\lvert S_i \rvert$
S_1	Depth	depth	FO4	9::3::36	10
S_2	Width	width L/S reorder queue store queue functional units	insn b/w entries entries count	4,8,16 15::15::45 14::14::42 1,2,4	3
S_3	Physical Registers	general purpose (GP) floating-point (FP) special purpose (SP)	count count count	40::10::130 40::8::112 42::6::96	10
S_4	Reservation Stations	branch fixed-point/memory floating-point	entries entries entries	6::1::15 10::2::28 5::1::14	10
S_5	I-L1 Cache	i-L1 cache size	log_2(entries)	7::1::11	5
S_6	D-L1 Cache	d-L1 cache size	log_2(entries)	6::1::10	5
S_7	L2 Cache	L2 cache size L2 cache latency	log_2(entries) cycles	11::1::15 6::2::14	5

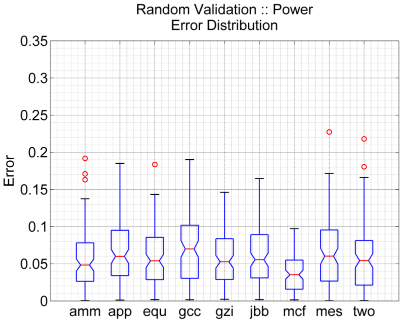
- Parameter space of 375,000 design points

Validation Approach

- **Framework**
  - Formulate models with 1,000 samples
  - Obtain 100 additional random samples for validation
  - Quantify percentage error, $100 * |\hat{y}_i - y_i|/y_i$
- **Comparison**
  - Simulator-reported performance, power
  - Regression-predicted performance, power

Prediction Accuracy

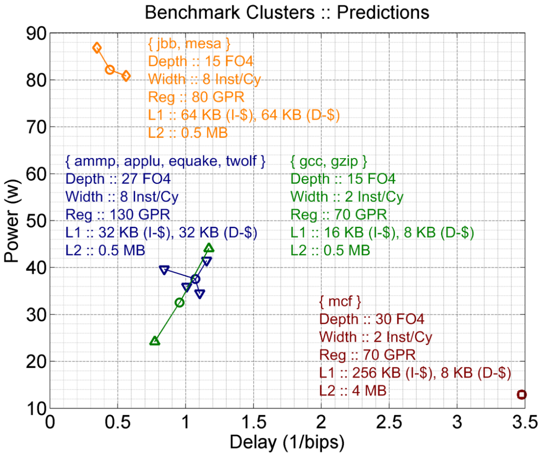




Multiprocessor Heterogeneity I

- **Motivation**
  - Evaluate trends in chip multiprocessor design
  - Mitigate penalties from single design compromise
- **Objective**
  - Identify efficient heterogeneous design compromises
- **Approach**
  - Simulate 1K samples from design space
  - Formulate regression models for performance, power
  - Identify per benchmark optima ($bips^3/w$) via regression
  - Identify compromises via K-means clustering

Multiprocessor Heterogeneity II



Outline

Platforms and Workloads

- **Platforms**
  - Blue Gene/L
  - Intel Xeon Clusters (ALC,MCR)
- **Numerical Methods**
  - Semicoarsening Multigrid 2000 (SMG2k)
  - High-Performance Linpack (HPL)
- **Statistical Framework**
  - R :: software environment for statistical computing
  - Hmisc and Design packages

Predictors :: Numerical Methods

- **Semicoarsening Multigrid 2000**

	Set	Parameters	Measure	Range	$\lvert S_i \rvert$
S_1	N_x	x-dim working set	grid points	10:20:510	26
S_2	N_y	y-dim		10:20:510	26
S_3	N_z	z-dim		10:20:510	26
S_4	P_x	x-dim processors	processors	1,8,64,512	4
S_5	P_y	y-dim		1,8,64,512	4
S_6	P_z	z-dim		1,8,64,512	4

  - Parameter space of $\sim$ 280,000 combinations

- **High-Performance Linpack**

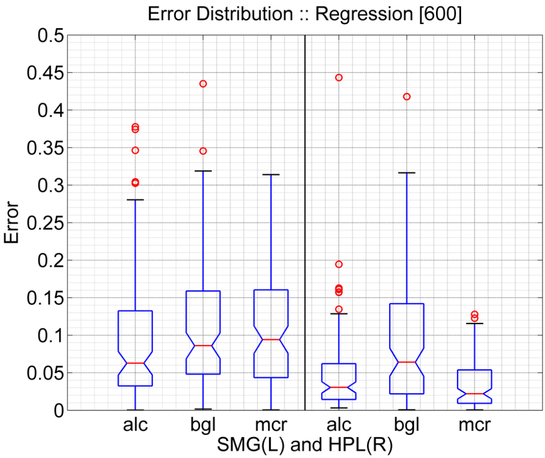
	Set	Parameters	Measure	Range	$\lvert S_i \rvert$
S_1	Matrix Size	N	sq. matrix dim	1000	1
S_2	Block Size	NB	sq. block dim	10:10:80	8
S_3	Processor Distribution	rows (P) columns (Q)	log_2(procs) log_2(procs)	0:1:9 9-P	10
S_4	Panel Factor	$PFACT$	algorithm	L,R,C	3
S_5	Recursive Factor	$RFACT$	algorithm	L,R,C	3
S_6	Recursive Base	$NBMIN$	block size	1:1:8	8
S_7	Recursive Sub-Panels	$NDIV$	sub-panels	2:1:4	3
S_8	Broadcast	$BCAST$	algorithm	1rg, 1rM, 2rg, 2rM, Lng, LnM	6

  - Parameter space of $\sim$ 100,000 combinations

Validation Approach

- **Framework**
  - Formulate models with 600 samples
  - Obtain 100 additional random samples for validation
  - Quantify percentage error, $100 * |\hat{y}_i - y_i|/y_i$
- **Comparison**
  - Profiled execution time
  - Regression-predicted execution time

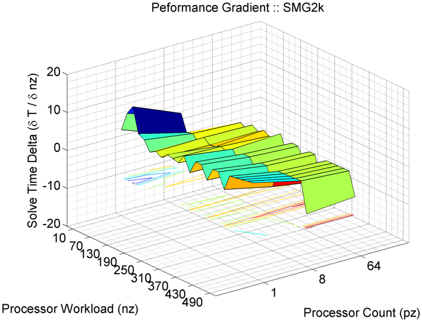
Prediction Accuracy

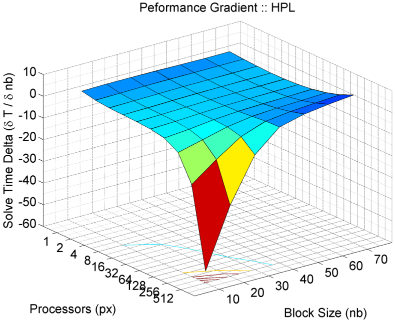


Performance Gradients I

- **Motivation**
  - Model performance empirically
  - Circumvent analytical complexity
- **Objective**
  - Understand performance topology, bottlenecks
- **Approach**
  - Measure 600 samples from parameter space
  - Formulate regression models for performance
  - Predict execution time for every point
  - Compute numerical gradients with local differences

Performance Gradients II





Outline

Conclusion

- **Exploration Paradigm**
  - Comprehensively understand parameter space
  - Selectively measure modest number of points
  - Efficiently leverage measured data with inference
- **Model Evaluation**
  - 7.2%, 5.4% median errors for μ-arch performance, power
  - 5.1%, 3.1% median errors for SMG2K, HPL performance
- **Future Directions**
  - Chip multiprocessors and on-chip interconnect
  - Additional applications and compiler parameters
  - Combine microarchitecture, application models

Further Reading

www.deas.harvard.edu/∼bclee

B.C. Lee and D.M. Brooks and B.R. de Supinski and M. Schulz and K. Singh and S.A. McKee.
Methods of inference & learning for performance modeling of parallel applications.
PPoPP'07: Symposium on Principles & Practice of Parallel Programming, March 2007.

B.C. Lee and D.M. Brooks.
Illustrative design space studies with microarchitectural regression models.
HPCA-13: International Symposium on High Performance Computer Architecture, Feb 2007.

B.C. Lee and D.M. Brooks.
Accurate, efficient regression modeling for microarchitectural performance, power prediction.
ASPLOS-XII: International Conference on Architectural Support for Programming Languages & Operating Systems, Oct 2006.

B.C. Lee and M. Schulz and B. de Supinski.
Regression strategies for parameter space exploration: A case study in semicoarsening multigrid & R.
Technical Report UCRL-TR-224851, Lawrence Livermore National Laboratory, Sept 2006.

B.C. Lee and D.M. Brooks.
Statistically rigorous regression modeling for the microprocessor design space.
MoBS-2: Workshop on Modeling, Benchmarking, & Simulation, June 2006.

B.C. Lee and D.M. Brooks.
Regression modeling strategies for microarchitectural performance & power prediction.
Harvard University Technical Report TR-08-06, March 2006.

References I

- Y. Li, B.C. Lee, D. Brooks, Z. Hu, K. Skadron.
  CMP design space exploration subject to physical constraints.
  HPCA-12: International Symposium on High-Performance Computer Architecture, Feb 2006.
- L. Eeckhout, S. Nussbaum, J. Smith, and K. DeBosschere.
  Statistical simulation: Adding efficiency to the computer designer's toolbox.
  IEEE Micro, Sept/Oct 2003.
- R. Liu and K. Asanovic.
  Accelerating architectural exploration using canonical instruction segments.
  In *International Symposium on Performance Analysis of Systems and Software*, Austin, Texas, March 2006.
- T. Sherwood, E. Perelman, G. Hamerly, and B. Calder.
  Automatically characterizing large scale program behavior.
  ASPLOX-X: Architectural Support for Programming Languages and Operating Systems, October 2002.
- B.C. Lee and D.M. Brooks.
  Effects of pipeline complexity on SMT/CMP power-performance efficiency.
  ISCA-32: Workshop on Complexity Effective Design, June 2005.

References II

C. Stone.
Comment: Generalized additive models.
Statistical Science, 1986.

F. Harrell.
Regression modeling strategies.
Springer, New York, NY, 2001.

J. Yi, D. Lilja, and D. Hawkins.
Improving computer architecture simulation methodology by adding statistical rigor.
IEEE Computer, Nov 2005.

P. Joseph, K. Vaswani, and M. J. Thazhuthaveetil.
Construction and use of linear regression models for processor performance analysis.
In *Proceedings of the 12th Symposium on High Performance Computer Architecture*, Austin, Texas, February 2006.

S. Nussbaum and J. Smith.
Modeling superscalar processors via statistical simulation.
In *PACT2001: International Conference on Parallel Architectures and Compilation Techniques*, Barcelona, Sept 2001.

Treatment of Missing Data

- **Missing Completely at Random (MCAR)**
  - Treat unobserved design points as missing data
  - Sampling UAR ensures observations are MCAR
  - Data is missing for reasons unrelated to characteristics or responses of the configuration
- **Informative Missing**
  - Data is more likely missing if their responses are systematically higher or lower
  - “Missingness” is non-ignorable and must also be modeled
  - Sampling UAR avoids such modeling complications

Predictor Non-Linearity I

- **Polynomial Transformations**
  - Undesirable peaks and valleys
  - Differing trends across regions
- **Linear Splines**
  - Piecewise linear regions separated by knots
  - Inadequate for complex, highly curved relationships
- **Restricted Cubic Splines**
  - Higher order polynomials provide better fits
  - Continuous at knots
  - Linear constraint on tails

Predictor Non-Linearity II

- **Location of Knots**
  - Location of knots less important than number of knots
  - Place knots at fixed predictor quantiles
- **Number of Knots**
  - Flexibility, risk of over-fitting increases with knot count
  - 5 knots or fewer are often sufficient [1]
  - 4 knots is a good compromise between flexibility, over-fitting
  - Fewer knots required for small data sets

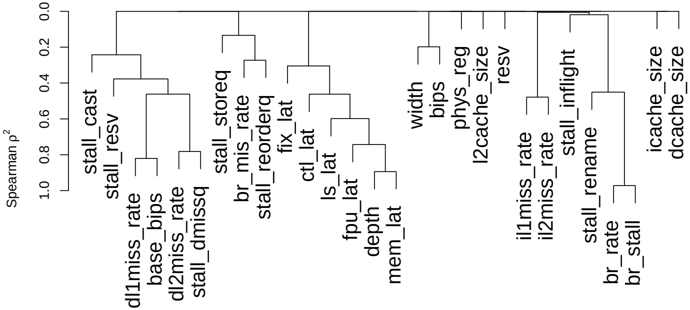
[1] Stone [SS'86]

Derivation Overview

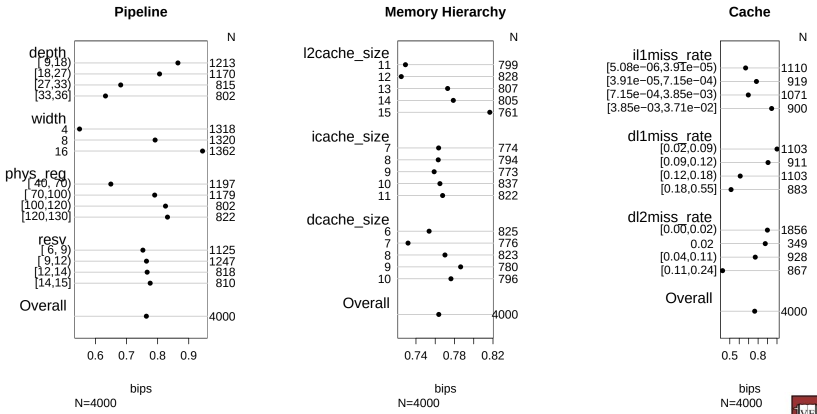
- **Spatial Sampling**
- **Hierarchical Clustering**
- **Association Analysis**
  - qualitative scatterplots, quantitative ρ^2
- **Model Specification**
  - predictor interaction, non-linearity
- **Assessing Fit**
  - R^2 statistic
- **Residual Analysis**
  - normality (quantile-quantile), randomness (scatterplots)
- **Significance Testing**
  - hypothesis testing, F-statistic, p-values

[4] Lee[ASPLOS'06], Lee[PPoPP'07]

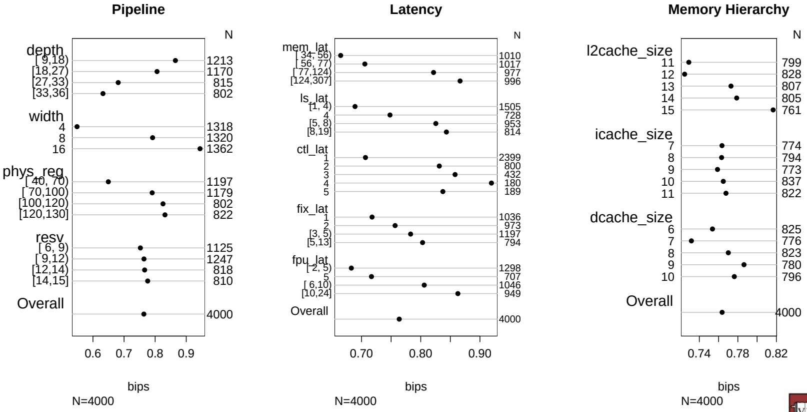
Variable Clustering



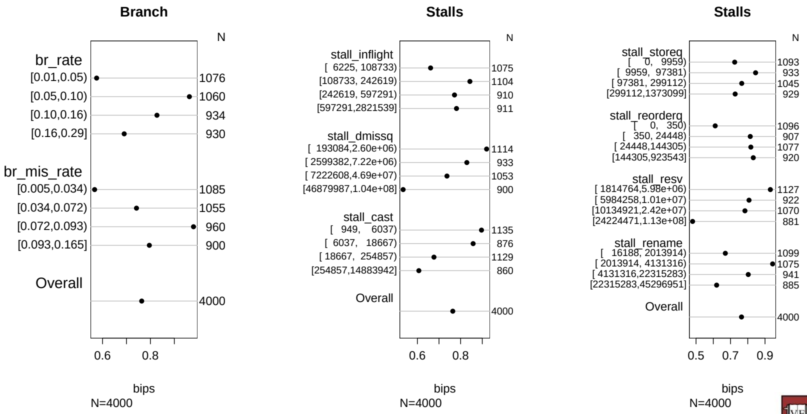
Performance Associations



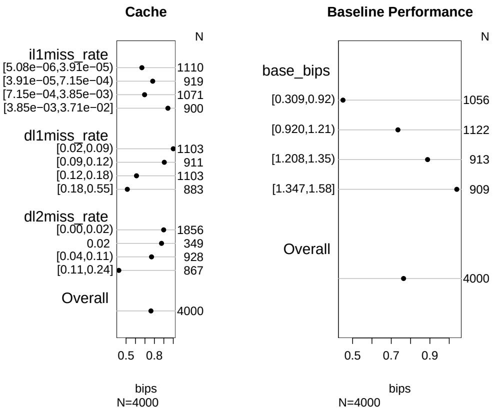
Performance Associations I



Performance Associations II



Performance Associations III



Assessing Fit

- **Multiple Correlation Statistic**
  - R^2 is fraction of response variance captured by predictors
  - Large R^2 suggests better fit to observed data
  - $R^2 \rightarrow 1$ suggests over-fitting (less likely if $p < n/20$)

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \frac{1}{n}\sum_{i=1}^{n} y_i)^2}$$

- **Residual Distribution Assumptions**
  - Residuals are normally distributed, $e_i \sim N(0, \sigma^2)$
  - No correlation between residuals and response, predictors
  - Validate by scatterplots and quantile-quantile plots

$$\hat{e}_i = y_i - \hat{\beta}_0 - \sum_{j=0}^{p} \hat{\beta}_j x_{ij}$$

Significance Testing I

- **Approach**
  - Given two nested models, hypothesis H_0 states additional predictors in larger model have no response association
  - Test H_0 with F-statistics and p-values
- **Example**
  - Predictor interaction requires comparing nested models
  - Consider a model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_1 x_2$.
  - Test significance of x_1 with null hypothesis $H_0 : \beta_1 = \beta_3 = 0$

Significance Testing II

- **F-Statistic**
  - Compare two nested models using their R^2 and F-statistic
  - R^2 is fraction of response variance captured by predictors

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \frac{1}{n}\sum_{i=1}^{n} y_i)^2}$$

  - F-statistic of two nested models follows F distribution

$$F_{k,n-p-1} = \frac{R^2 - R_*^2}{k} \times \frac{n-p-1}{1-R^2}$$

- **P-Values**
  - Probability F-statistic greater than or equal to observed value would occur under H_0
  - Small p-values cast doubt on H_0

Significance Testing IV

- **Microarchitectural Predictors**
  - Majority of F-tests imply significance (p-values $< 2.2E-16$)
  - Several predictors were less significant
    - Control latency (p-value $= 0.1247$)
    - Reservation station size (p-value $= 0.1239$)
    - L1 instruction cache size (p-value $= 0.02941$)
- **Application-Specific Predictors**
  - Majority of F-tests imply significance (p-values $< 2.2E-16$)
  - Pipeline stalls classified by structure are less significant
    - Completion and reorder queue stalls (p-values > 0.4)

Related Work

- **Statistical Significance Ranking**
  - Yi :: Plackett-Burman, effect rankings
  - Joseph :: Stepwise regression, coefficient rankings
  - Bound parameter values to improve tractability
  - Require simulation for estimation
- **Synthetic Workloads**
  - Eeckhout :: Profile workloads to obtain synthetic traces
  - Nussbaum :: Superscalar and SMP simulation
  - Obtain distribution of instructions and data dependencies
  - Require simulation with smaller traces for estimation