Performance Prediction of NUMA Placement
A Machine-Learning Approach

Fanourios Arapidis, Vasileios Karakostas, Nikela Papadopoulou, Konstantinos Nikas, Georgios Goumas, Nectarios Koziris

Computing Systems Laboratory
School of Electrical and Computer Engineering
National Technical University of Athens, ICCS

December 2018

Introduction

NUMA systems

- Consist of multiple sockets and memory modules
- Controled by a single OS/hypervisor

Introduction

NUMA systems

- Consist of multiple sockets and memory modules
- Controled by a single OS/hypervisor

Advantages

- Provide large amounts of shared memory
- Provide necessary infrastructure for scale-up workloads
- Opportunity to collocate multiple workloads and increase utilization

Introduction

NUMA systems

- Consist of multiple sockets and memory modules
- Controled by a single OS/hypervisor

Advantages

- Provide large amounts of shared memory
- Provide necessary infrastructure for scale-up workloads
- Opportunity to collocate multiple workloads and increase utilization

- Placement of threads and memory matters

Introduction

NUMA systems

- Consist of multiple sockets and memory modules
- Controled by a single OS/hypervisor

Advantages

- Provide large amounts of shared memory
- Provide necessary infrastructure for scale-up workloads
- Opportunity to collocate multiple workloads and increase utilization

- Placement of threads and memory matters
- Common wisdom: Memory locality is the best policy

Introduction

NUMA systems

- Consist of multiple sockets and memory modules
- Controled by a single OS/hypervisor

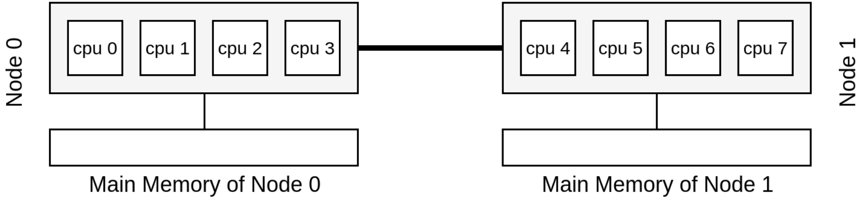
Advantages

- Provide large amounts of shared memory
- Provide necessary infrastructure for scale-up workloads
- Opportunity to collocate multiple workloads and increase utilization

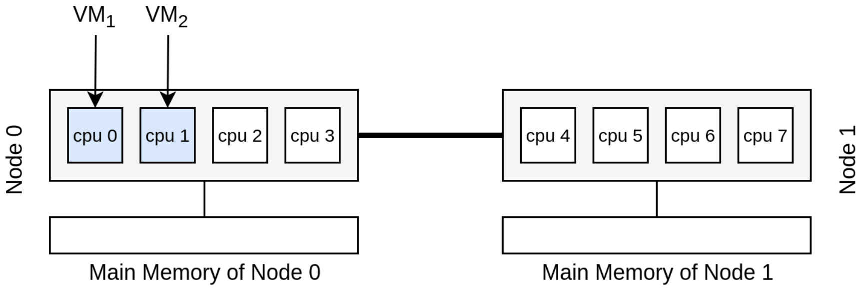
- Placement of threads and memory matters
- Common wisdom: Memory locality is the best policy
- But not always feasible

Problematic Scenario

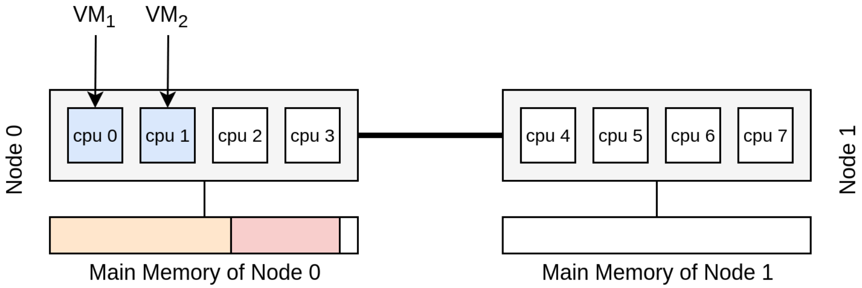
Problematic Scenario



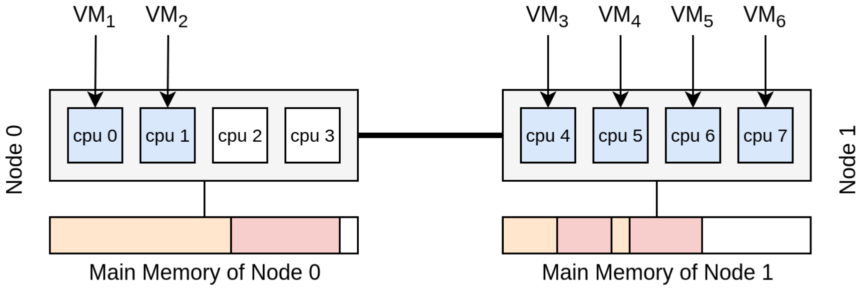
Problematic Scenario



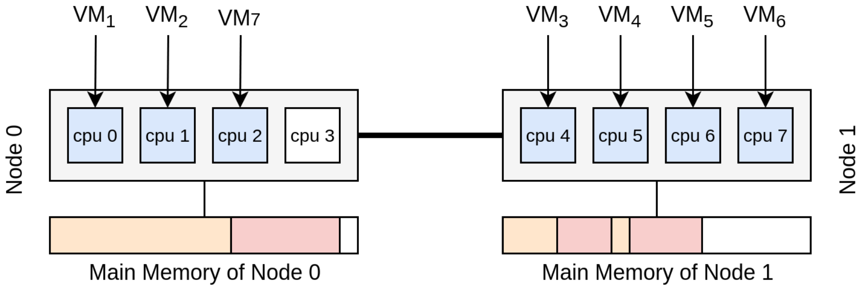
Problematic Scenario



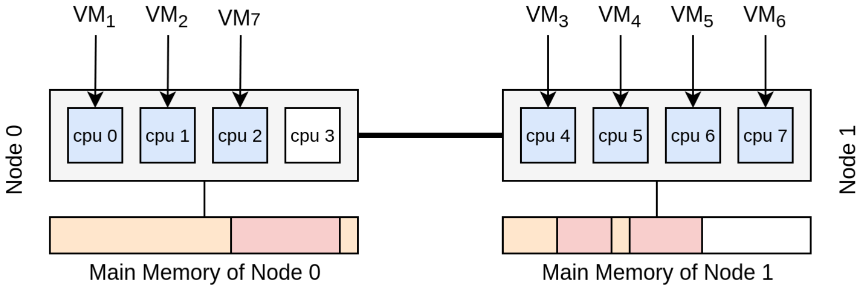
Problematic Scenario



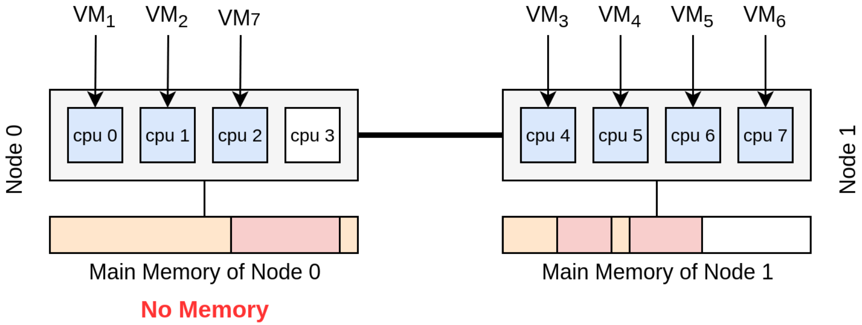
Problematic Scenario



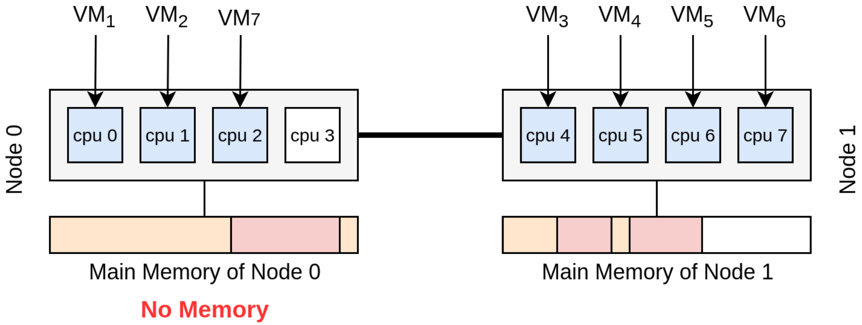
Problematic Scenario



Problematic Scenario



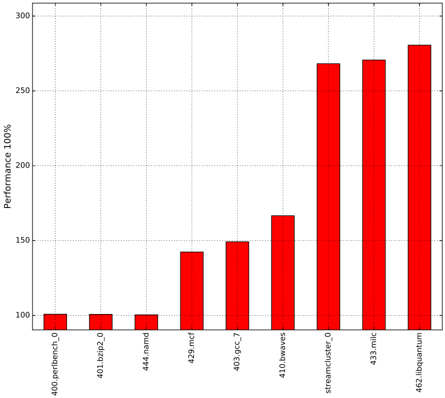
Problematic Scenario



Which VM to move?

Motivation

Not all applications are affected in the same way by the placement of core and memory



In a nutshell

We want to predict the impact on performance of core and memory placement in non-uniform memory access (NUMA) systems

In a nutshell

We want to predict the impact on performance of core and memory placement in non-uniform memory access (NUMA) systems

The performance depends on

- Architecture
- Application's characteristics

In a nutshell

We want to predict the impact on performance of core and memory placement in non-uniform memory access (NUMA) systems

The performance depends on

- Architecture
- Application's characteristics

Our Approach

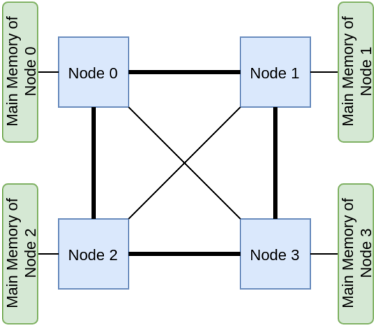
- Measure parameters from applications
- Model performance using non-linear functions
- Train the functions with machine learning techniques
- Evaluate the predictions

Outline

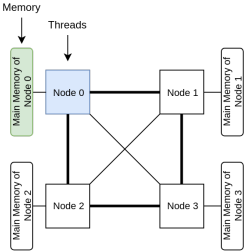
Platform Details

Characteristics

- 4 nodes
- Intel Xeon E5-4620, 2.2GHz
- Sandy Bridge
- 8 cores per node (2 threads)
- 64GB RAM per node
- Cache
  - L1 32KB, shared per core
  - L2 256KB, shared per core
  - L3 (LLC) 16MB, shared per NUMA node

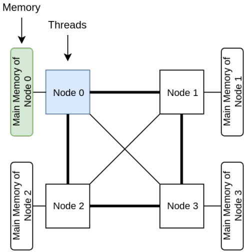


Placement Scenarios

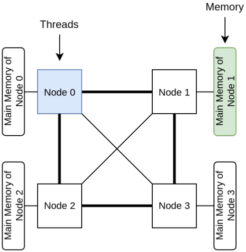


(a) Local

Placement Scenarios

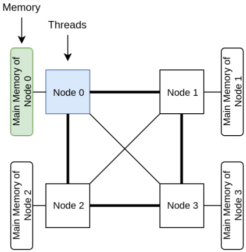


(a) Local

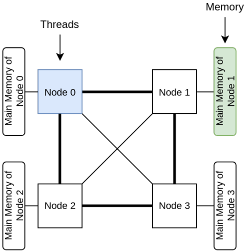


(b) Remote near

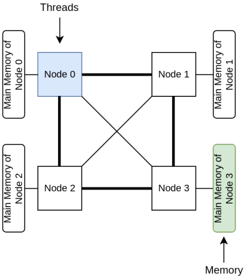
Placement Scenarios



(a) Local



(b) Remote near



(c) Remote far

Tools

numactl

- Provides necessary information about the system (nodes, processors)
- Allows to perform interleave and membind (remote) executions

perf

- Measures the parameters of our applications by using the performance counters

Benchmarking

- **Spec2006**

- **Parsec**

Benchmarking

- **Spec2006**
  - 26 different benchmarks
  - Multiple input files
  - A total of 52 different executions
- **Parsec**

Benchmarking

- **Spec2006**
  - 26 different benchmarks
  - Multiple input files
  - A total of 52 different executions
- **Parsec**
  - 12 different benchmarks
  - Single input files
  - A total of 12 different executions

Benchmarking

- **Spec2006**
  - 26 different benchmarks
  - Multiple input files
  - A total of 52 different executions
- **Parsec**
  - 12 different benchmarks
  - Single input files
  - A total of 12 different executions
- In total, we have 64 points for each topology

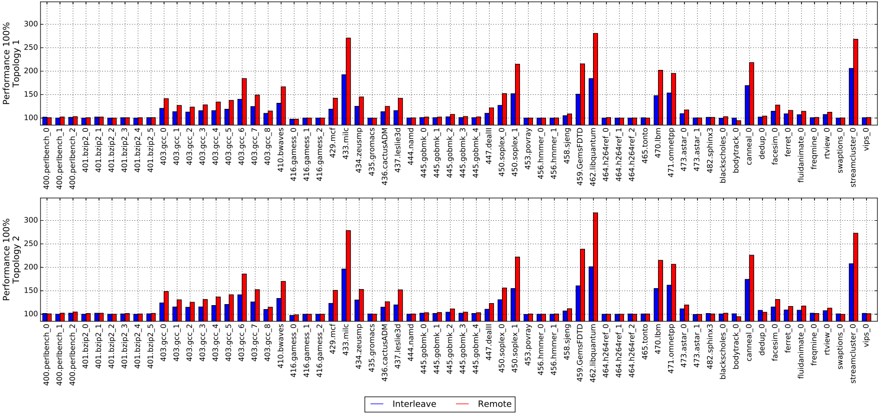
Benchmarking

- **Spec2006**
  - 26 different benchmarks
  - Multiple input files
  - A total of 52 different executions
- **Parsec**
  - 12 different benchmarks
  - Single input files
  - A total of 12 different executions
- In total, we have 64 points for each topology

Performance

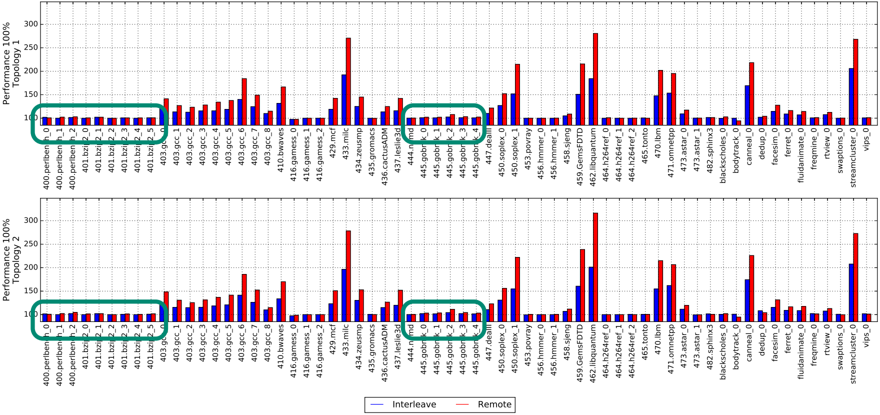
- $\text{Performance}(i,j) = \frac{\text{Total Time}(i,j)}{\text{Total Time}_{local}} \cdot 100\%$

Performance



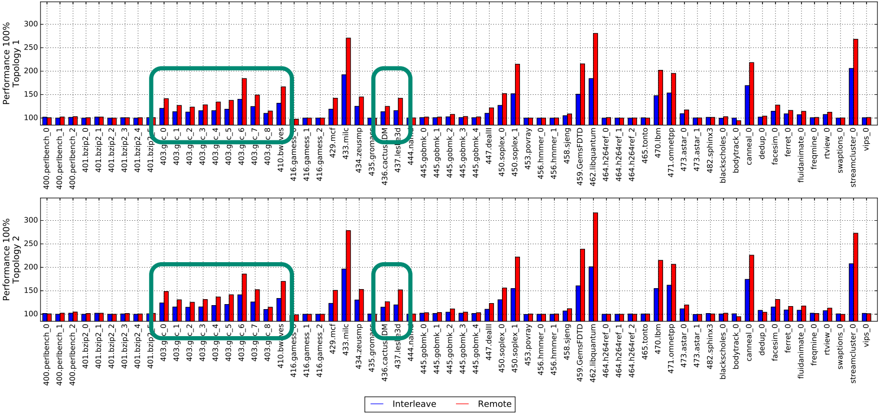
Impact of NUMA placement on performance

Performance



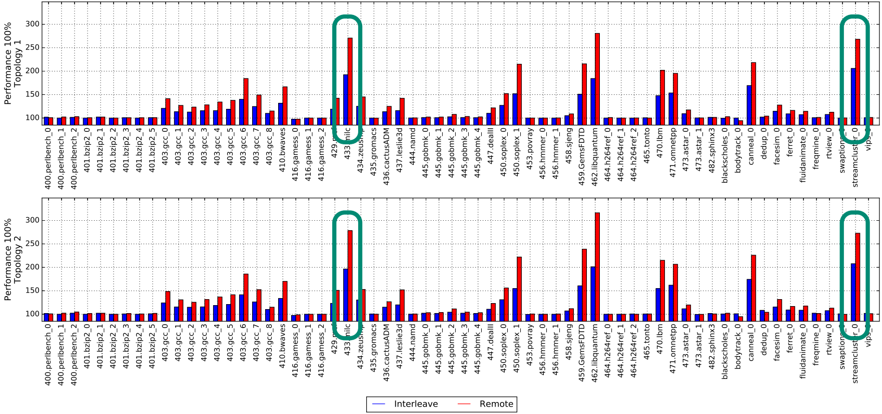
1. Low impact to performance

Performance



2. Medium impact to performance

Performance



3. High impact to performance

Application Parameters

1. Last level Cache Misses per Kilo Instructions

$$\text{MPKI} = \frac{\sum_{i=1}^{t} \text{LLC-load-misses[i]} + \sum_{i=1}^{t} \text{LLC-store-misses[i]}}{\sum_{i=1}^{t} \text{instructions[i]}} \cdot 1000$$

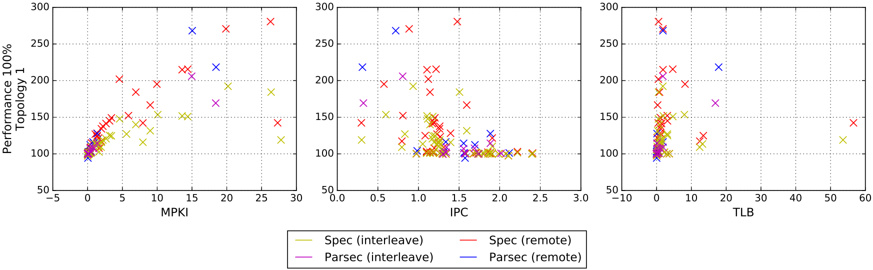
2. Instructions per Cycle

$$\text{IPC} = \frac{\sum_{i=1}^{t} \text{instructions[i]}}{\sum_{i=1}^{t} \text{cpu-cycles[i]}}$$

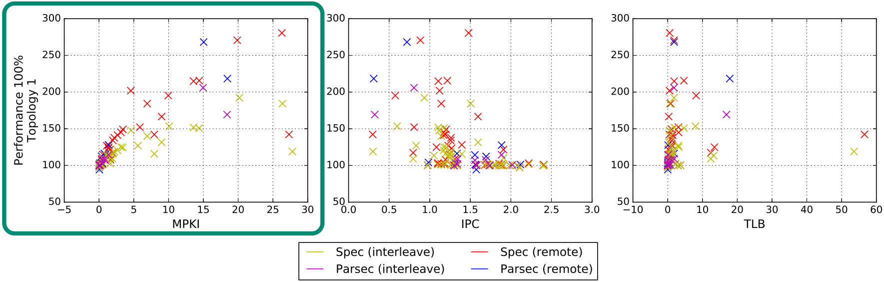
3. TLB Misses per Kilo Instructions

$$\text{TLB} = \frac{\sum_{i=1}^{t} \text{dTLB-load-misses[i]} + \sum_{i=1}^{t} \text{dTLB-store-misses[i]}}{\sum_{i=1}^{t} \text{instructions[i]}} \cdot 1000$$

Correlating Parameters with Performance

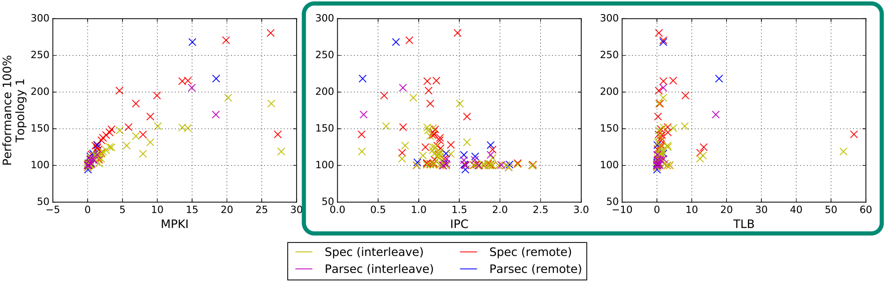


Correlating Parameters with Performance



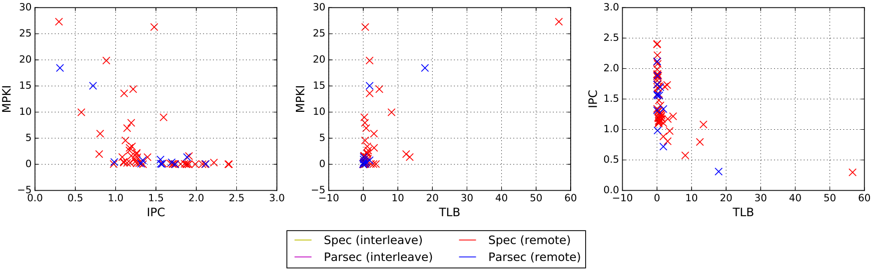
Approach with logarithmic function

Correlating Parameters with Performance

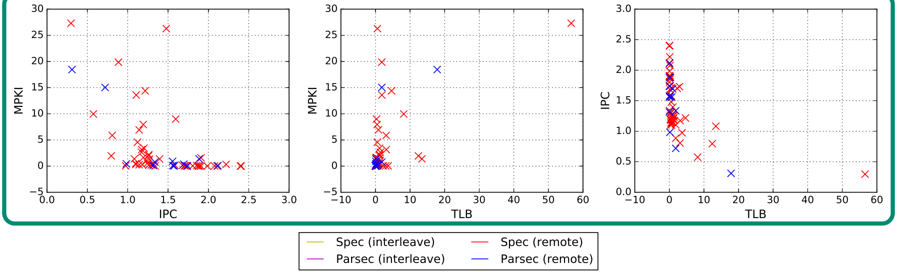


Approach with non-linear function

Correlations between Parameters



Correlations between Parameters



Approach with non-linear function

Generating Candidate Models

The terms are divided into

1. Basic
2. Correlations

Generating Candidate Models

The terms are divided into

1. Basic
    1. Dominant
    2. Secondary
2. Correlations

Generating Candidate Models

The terms are divided into

1. Basic
   1. Dominant
   2. Secondary
2. Correlations

Dominant term

- $a_j X_i^{n_1}$
- $a_j log(X_i + 1)^{n_1}$
- $a_j log(X_i + 1)^{n_1} + a_{j+1} X_i^{n_2}$

Secondary term

- $a_j X_i^{n_1}$

Size	Parameters
1	MPKI
	IPC
	TLB
2	MPKI, IPC
	MPKI, TLB
	IPC, TLB
3	MPKI, IPC, TLB

Including High-order Terms

Correlations

- Rank
  Maximum number of different parameters

Including High-order Terms

Correlations

- Rank
  Maximum number of different parameters
- $a_i(\prod_{j=1}^{\text{rank}} X_j^{n_{k+j-1}})$

Including High-order Terms

Correlations

- Rank
  Maximum number of different parameters
- $a_i(\prod_{j=1}^{rank} X_j^{n_{k+j-1}})$

Rank	Correlations
2	$a_i(X_1^{n_j} \cdot X_2^{n_{j+1}})$
	$a_i(X_1^{n_j} \cdot X_3^{n_{j+1}})$
	$a_i(X_2^{n_j} \cdot X_3^{n_{j+1}})$
3	$a_i(X_1^{n_j} \cdot X_2^{n_{j+1}} \cdot X_3^{n_{j+2}})$

Including High-order Terms

Correlations

- Rank
  Maximum number of different parameters
- $a_i(\prod_{j=1}^{\text{rank}} X_j^{n_{k+j-1}})$

Rank	Correlations
2	$a_i(X_1^{n_j} \cdot X_2^{n_{j+1}})$
	$a_i(X_1^{n_j} \cdot X_3^{n_{j+1}})$
	$a_i(X_2^{n_j} \cdot X_3^{n_{j+1}})$
3	$a_i(X_1^{n_j} \cdot X_2^{n_{j+1}} \cdot X_3^{n_{j+2}})$

We automatically generate models following a model generation approach similar to "Fast multi-parameter performance modeling" *[CLUSTER'16]*

Model Training

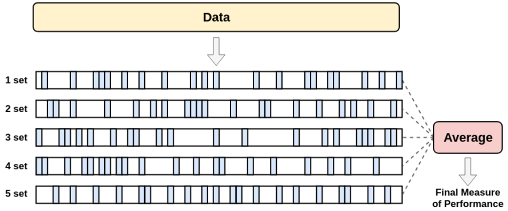
To train and evaluate the accuracy of the generated prediction models

Model Training

To train and evaluate the accuracy of the generated prediction models

Cross Validation

- Random subsampling validation
- Repeated 5 times
- Randomly generated sets (44 train, 20 test)
- Sets may overlap
- Final score: average of all test sets



Modeling

Modeling stages

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation
3. Model Training

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation
3. Model Training
4. Model Scoring

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation
3. Model Training
4. Model Scoring

Metrics

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation
3. Model Training
4. Model Scoring

Metrics

1. R^2 (coefficient of determination)
   Best possible score is 1.0 and it can be negative

Modeling

Modeling stages

1. Data collection
   {Performance (i, j), MPKI(local), IPC(local), TLB(local)}
2. Model generation
3. Model Training
4. Model Scoring

Metrics

1. R^2 (coefficient of determination)
   Best possible score is 1.0 and it can be negative
2. MAE (mean absolute error)
   Best possible score is 0

Results

Top	Function	R^2	MAE
1	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_1^{n_2} + a_3 \cdot x_3 + b$	0.9311	5.3154
2	$a_1 \cdot x_1^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9303	5.3562
3	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2^{n_1} + a_4 \cdot x_3 + b$	0.9291	5.4005
4	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1 + a_3 \cdot x_3 + a_4 \cdot (x_2^{n_1} \cdot x_1^{n_2}) + b$	0.9283	5.3818
5	$a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_1} \cdot x_2^{n_2}) + b$	0.9278	5.4170
6	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_1 + a_3 \cdot x_2^{n_2} + a_4 \cdot x_3 + b$	0.9275	5.4801
7	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9254	5.6392
8	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_1^{n_2} + a_4 \cdot x_3 + b$	0.9254	5.3471
9	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9246	5.6528
10	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_3 + b$	0.9236	5.8177
11	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9234	5.8351
12	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_2} \cdot x_2^{n_3}) + b$	0.9229	5.6140

Results

Top	Function	R^2	MAE
1	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_1^{n_2} + a_3 \cdot x_3 + b$	0.9311	5.3154
2	$a_1 \cdot x_1^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9303	5.3562
3	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2^{n_1} + a_4 \cdot x_3 + b$	0.9291	5.4005
4	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1 + a_3 \cdot x_3 + a_4 \cdot (x_2^{n_1} \cdot x_1^{n_2}) + b$	0.9283	5.3818
5	$a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_1} \cdot x_2^{n_2}) + b$	0.9278	5.4170
6	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_1 + a_3 \cdot x_2^{n_2} + a_4 \cdot x_3 + b$	0.9275	5.4801
7	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9254	5.6392
8	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_1^{n_2} + a_4 \cdot x_3 + b$	0.9254	5.3471
9	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9246	5.6528
10	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_3 + b$	0.9236	5.8177
11	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9234	5.8351
12	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_2} \cdot x_2^{n_3}) + b$	0.9229	5.6140

Results

Top	Function	R^2	MAE
1	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_1^{n_2} + a_3 \cdot x_3 + b$	0.9311	5.3154
2	$a_1 \cdot x_1^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9303	5.3562
3	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2^{n_1} + a_4 \cdot x_3 + b$	0.9291	5.4005
4	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1 + a_3 \cdot x_3 + a_4 \cdot (x_2^{n_1} \cdot x_1^{n_2}) + b$	0.9283	5.3818
5	$a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_1} \cdot x_2^{n_2}) + b$	0.9278	5.4170
6	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_1 + a_3 \cdot x_2^{n_2} + a_4 \cdot x_3 + b$	0.9275	5.4801
7	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9254	5.6392
8	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_1^{n_2} + a_4 \cdot x_3 + b$	0.9254	5.3471
9	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9246	5.6528
10	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_3 + b$	0.9236	5.8177
11	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9234	5.8351
12	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_2} \cdot x_2^{n_3}) + b$	0.9229	5.6140

Results

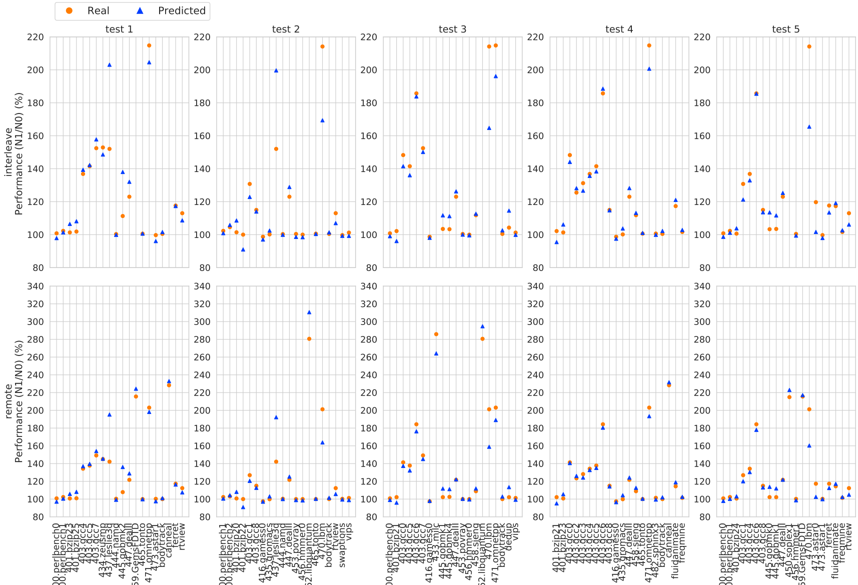
Top	Function	R^2	MAE
1	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_1^{n_2} + a_3 \cdot x_3 + b$	0.9311	5.3154
2	$a_1 \cdot x_1^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9303	5.3562
3	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2^{n_1} + a_4 \cdot x_3 + b$	0.9291	5.4005
4	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1 + a_3 \cdot x_3 + a_4 \cdot (x_2^{n_1} \cdot x_1^{n_2}) + b$	0.9283	5.3818
5	$a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_1} \cdot x_2^{n_2}) + b$	0.9278	5.4170
6	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_1 + a_3 \cdot x_2^{n_2} + a_4 \cdot x_3 + b$	0.9275	5.4801
7	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9254	5.6392
8	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_1^{n_2} + a_4 \cdot x_3 + b$	0.9254	5.3471
9	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9246	5.6528
10	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_3 + b$	0.9236	5.8177
11	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9234	5.8351
12	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_2} \cdot x_2^{n_3}) + b$	0.9229	5.6140

Results

Top	Function	R^2	MAE
1	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_1^{n_2} + a_3 \cdot x_3 + b$	0.9311	5.3154
2	$a_1 \cdot x_1^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9303	5.3562
3	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2^{n_1} + a_4 \cdot x_3 + b$	0.9291	5.4005
4	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1 + a_3 \cdot x_3 + a_4 \cdot (x_2^{n_1} \cdot x_1^{n_2}) + b$	0.9283	5.3818
5	$a_1 \cdot x_1 + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_1} \cdot x_2^{n_2}) + b$	0.9278	5.4170
6	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_1 + a_3 \cdot x_2^{n_2} + a_4 \cdot x_3 + b$	0.9275	5.4801
7	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1 + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9254	5.6392
8	$a_1 \cdot log(x_2+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_1^{n_2} + a_4 \cdot x_3 + b$	0.9254	5.3471
9	$a_1 \cdot log(x_1+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_2 + a_4 \cdot x_3 + b$	0.9246	5.6528
10	$a_1 \cdot log(x_2+1) + a_2 \cdot x_1^{n_1} + a_3 \cdot x_3 + b$	0.9236	5.8177
11	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2^{n_2} + a_3 \cdot x_3 + b$	0.9234	5.8351
12	$a_1 \cdot log(x_1+1)^{n_1} + a_2 \cdot x_2 + a_3 \cdot x_3 + a_4 \cdot (x_1^{n_2} \cdot x_2^{n_3}) + b$	0.9229	5.6140

Best function: $R^2 = 0.9311$ and MAE $= 5.3154$

Predictions using the Best Model



Challenges on Training

Problems

- We need to have at least as many measurement points as are the most volatile variables
- Training time is high due to the free variables n_i
- Many functions will fail in training
- The functions that would fail in training will take a long time

Solutions

- The free variables n_i are assigned discrete values within the range [-2, -1.5, -1, -0.5, 0.5, 1, 1.5, 2]
- The maximum number of terms is at most 7
- The correlation powers are the same as the basic terms

Conclusions and Future work

Conclusions

- Studied the performance with respect to NUMA placement
- Correlated the performance with MPKI, IPC, and TLB
- Built models that can predict performance with high accuracy

Conclusions and Future work

Conclusions

- Studied the performance with respect to NUMA placement
- Correlated the performance with MPKI, IPC, and TLB
- Built models that can predict performance with high accuracy

Future work

- Study larger scale NUMA systems
- Model different execution and placement scenarios
- Extend the model with information regarding machine status

Thank you

Fanourios Arapidis
farap@cslab.ece.ntua.gr