ScalPerf'23

Between ML and HPC: Where do (whitebox) performance models fit?

Y.C. Tay
National University of Singapore

Reflections and Perspectives on Avenues to Performance

(Reflection) What have the main novelties been during the past decade, in all aspects of HPC?

(Perspective) What is our educated guess on the key developments for the next decade?

1. Deployment of highly structured models, which significantly leverage domain-specific knowledge and insights coming from human intuition. Typically, data representations are developed that enable casting the problem in a homogeneous form. This paradigm encompasses most of the work that has been presented and discussed during the two decades of ScalPerf.

2. Deployment of traditional data regression models, which do not typically exploit domain specific knowledge. Examples are Support Vector Machines and Principal Component Analysis.

3. Deployment of recently developed AI/Machine-Learning models that enable data regularity to emerge, by massive computation on large data sets, with limited a priori domain-specific knowledge. Large Language Models are a crucial example, in very recent years. Experimentally, we see that patterns tend to emerge only at a rather large scale. Computationally, these models can be very expensive, with costs of the order of hundreds of millions of dollars. Avenues to reduce these costs are under active investigation.

Reflections and Perspectives on Avenues to Performance

(Reflection) What have the main novelties been during the past decade, in all aspects of HPC?

(Perspective) What is our educated guess on the key developments for the next decade?

1. Deployment of highly structured models, which significantly leverage domain-specific knowledge and insights coming from human intuition.

2. Deployment of traditional data regression models, which do not typically exploit domain specific knowledge.

3. Deployment of recently developed AI/Machine-Learning models that enable data regularity to emerge, by massive computation on large data sets, with limited a priori domain-specific knowledge.

1. Deployment of highly structured models, which significantly leverage domain-specific knowledge and insights coming from human intuition.

HPC for mathematical models (HiPC 2005):

The Potential of On-Chip Multiprocessing for QCD Machines*

Gianfranco Bilardi[1], Andrea Pietracaprina[1], Geppino Pucci[1], Fabio Schifano[2], and Raffaele Tripiccione[2]

Abstract. We explore the opportunities offered by current and forthcoming VLSI technologies to on-chip multiprocessing for Quantum Chromo Dynamics (QCD), a computational grand challenge for which over half a dozen specialized machines have been developed over the last two decades. Based on a careful study of the information exchange requirements of QCD both across the network and within the memory system, we derive the optimal partition of die area between storage and functional units. We show that a scalable chip organization holds the promise to deliver from hundreds to thousands flop per cycle as VLSI feature size scales down from 90 nm to 20 nm, over the next dozen years.

1. Deployment of highly structured models, which significantly leverage domain-specific knowledge and insights coming from human intuition.

mathematical models for HPC (HPCA2016):

Amdahl's Law for Lifetime Reliability Scaling in Heterogeneous Multicore Processors

William J. Song, Saibal Mukhopadhyay, and Sudhakar Yalamanchili

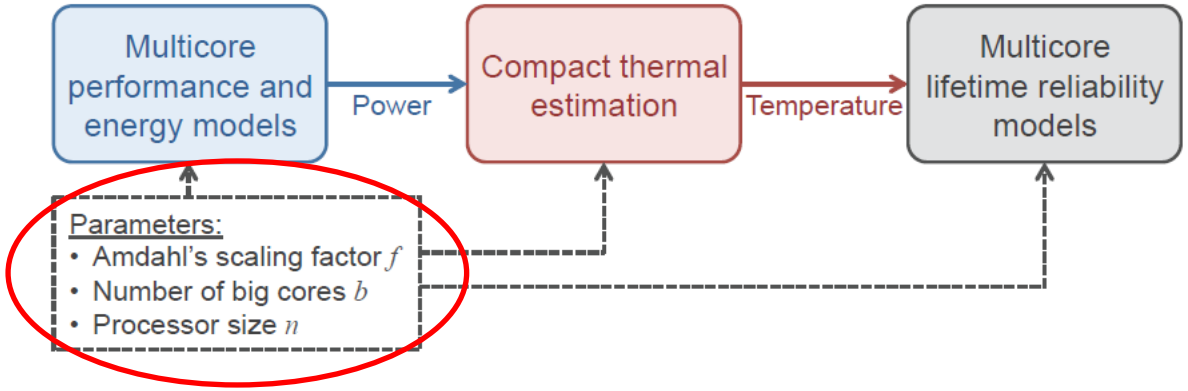


Figure 2: Modeling flow of performance, energy efficiency, thermal, and lifetime reliability characterization of heterogeneous multicore processors.

6.6 Heterogeneous Processor with Dynamic Scheduling

In a heterogeneous processor with dynamic scheduling, distinct types of cores are used to handle different phases of applications. By turning off unused cores, this scheduling policy benefits from improved reliability that is traded with performance degradation. The total failure rate of the processor is expressed as Eq. (23). The first term in this equation reflects the reliability impact of big cores during sequential operations, and the second term is the failure rate of small cores in parallel phases.

$$\lambda_{het:ds} = \frac{1-f}{s} \times \frac{\lambda_{b:seq}}{b} + \frac{f}{n - b \times r}(n - b \times r)\lambda_{s:par} \qquad (23)$$

2. Deployment of traditional data regression models, which do not typically exploit domain specific knowledge.

HPC for regression models (JPDC 2015):

J. Parallel Distrib. Comput. 76 (2015) 16–31

Contents lists available at ScienceDirect

J. Parallel Distrib. Comput.

journal homepage: www.elsevier.com/locate/jpdc

Scaling Support Vector Machines on modern HPC platforms

Yang You [a,b], Haohuan Fu [a,*], Shuaiwen Leon Song [c], Amanda Randles [d], Darren Kerbyson [c], Andres Marquez [c], Guangwen Yang [b], Adolfy Hoisie [c]

2. Deployment of traditional data regression models, which do not typically exploit domain specific knowledge.

regression models for HPC (SC 1999):

Adaptive Performance Prediction for Distributed Data-Intensive Applications*

Marcio Faerman† Alan Su† Richard Wolski‡ Francine Berman†

Abstract

The *computational grid* is becoming the platform of choice for large-scale distributed data-intensive applications. Accurately predicting the transfer times of remote data files, a fundamental component of such applications, is critical to achieving application performance. In this paper, we introduce a performance prediction method, **AdRM** (**Ad**aptive **R**egression **M**odeling), to determine file transfer times for network-bound distributed data-intensive applications.

3. Deployment of recently developed AI/Machine-Learning models that enable data regularity to emerge, by massive computation on large data sets, with limited a priori domain-specific knowledge.

HPC for big AI (HPCA 2023):

MPress: Democratizing Billion-Scale Model Training on Multi-GPU Servers via Memory-Saving Inter-Operator Parallelism

Quan Zhou[1], Haiquan Wang[1], Xiaoyan Yu[1], Cheng Li[1,2], Youhui Bai[1], Feng Yan[3], Yinlong Xu[1,2]

Reflections and Perspectives on Avenues to Performance

(Reflection) What have the main novelties been during the past decade, in all aspects of HPC?

(Perspective) What is our educated guess on the key developments for the next decade?

While pondering on the target questions, within the outlined framework,
it may be useful to identify the specific role of each layer of the computing stack,
such as VLSI and other technologies, circuit design, machine architecture,
compilers, performance tools, system software, application software,
programming models, algorithms, theory of computation, etc.
From which layers of the stack can we expect further performance
improvements in the coming decade?

HPC for big AI: where do performance models fit?

HPC for big AI: where do performance models fit?

ICML 2023: system (not algorithm)

FlexGen: High-Throughput Generative Inference of Large Language Models with a Single GPU

Ying Sheng [1] Lianmin Zheng [2] Binhang Yuan [3] Zhuohan Li [2] Max Ryabinin [4 5] Daniel Y. Fu [1] Zhiqiang Xie [1]
Beidi Chen [6 7] Clark Barrett [1] Joseph E. Gonzalez [2] Percy Liang [1] Christopher Ré [1] Ion Stoica [2] Ce Zhang [3]

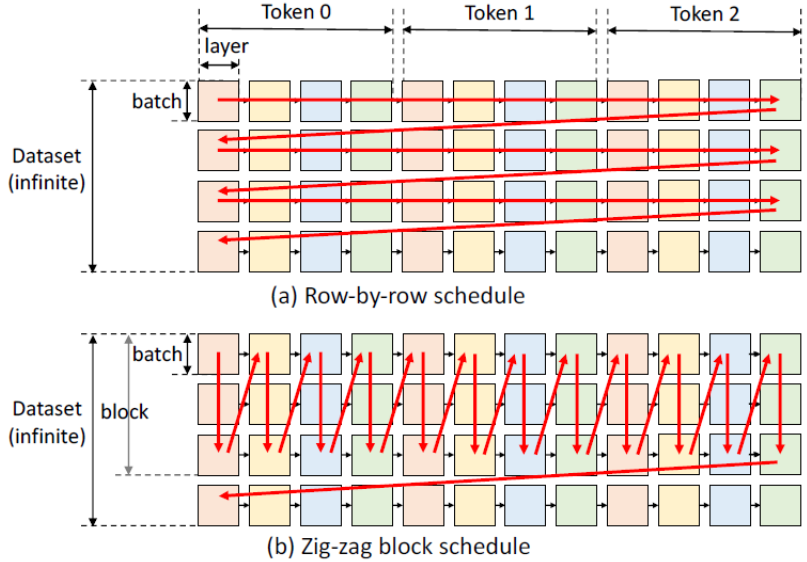


Figure 3. Two different schedules. The red arrows denote the computation order.

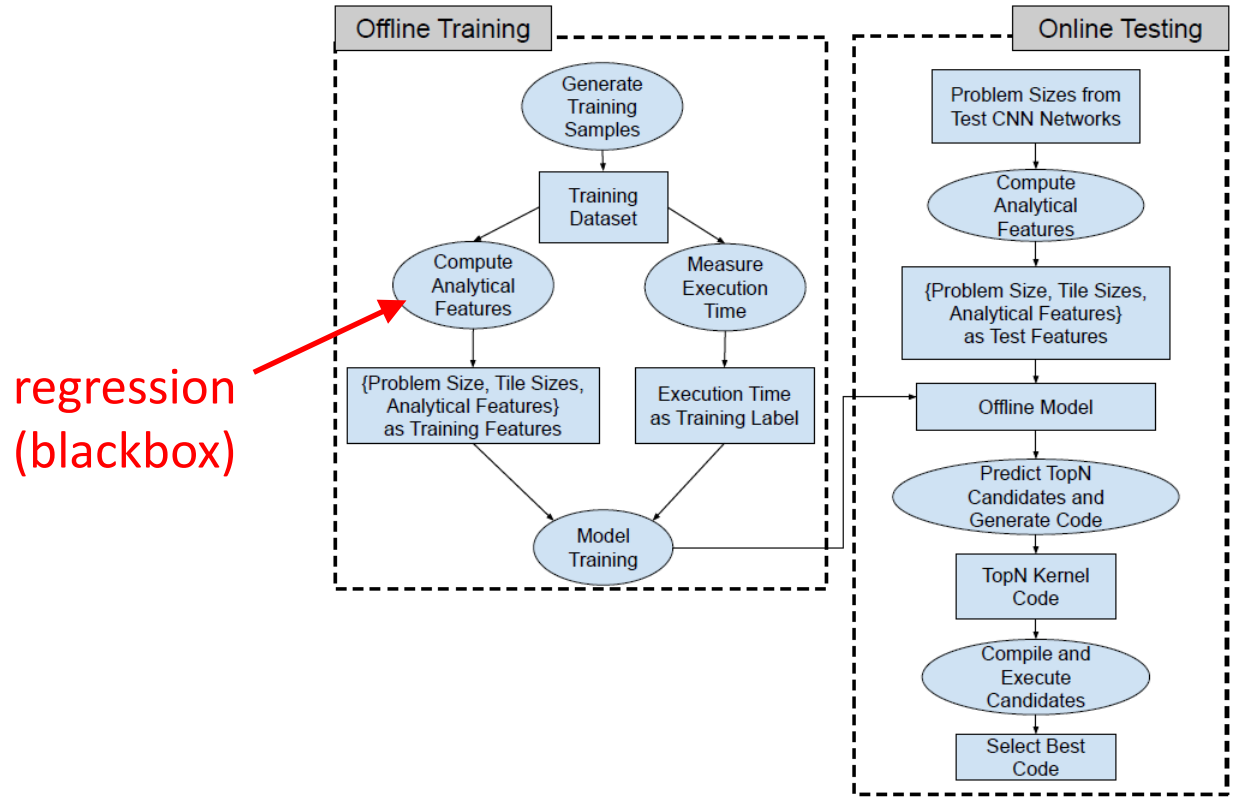
Theorem 4.1. *The I/O complexity of the zig-zag block schedule is within 2× of the optimal solution.*

HPC for big AI: where do performance models fit?

PACT 2022:

Effective Performance Modeling and Domain-Specific Compiler Optimization of CNNs for GPUs

Yufan Xu Qiwei Yuan Erik Curtis Barton



regression (blackbox)

Figure 5: Workflow of CNNOpt. Rectangular nodes represent data and oval nodes processes.

HPC for big AI: where do (whitebox) performance models fit?

AAAI 2022:

Reinforcement Learning for Datacenter Congestion Control

Chen Tessler[1 2*] **Yuval Shpigelman**[3] **Gal Dalal**[2] **Amit Mandelbaum**[3] **Doron Haritan Kazakov**[3]
Benjamin Fuhrer[3] **Gal Chechik**[2 4] **Shie Mannor**[1 2]

Reward. As the task is a multi-agent partially observable problem, the reward must be designed such that there exists a single fixed-point equilibrium.

Based on Appenzeller, Keslassy, and McKeown (2004), a good approximation of the RTT inflation (RTT-inflation $= \frac{\text{RTT}}{\text{base-RTT}}$) in a bursty system, where all flows transmit at the ideal rate, behaves like $\sqrt{N}$, where N is the number of flows. In this case, the combined transmission rate of all flows saturates the congestion point, the system is on the verge of congestion, and the major latency increase is due to the packets waiting in the congestion point's buffer. This latency is orders of magnitude higher than the empty-system routing latency. As such, we can assume that all flows sharing a congested path will observe a similar RTT inflation. We define

$$r_t = -\left(\textbf{target} - \frac{\text{RTT}_t^i}{\text{base-RTT}^i} \cdot \sqrt{\text{rate}_t^i}\right)^2, \qquad (1)$$

where **target** is a constant value shared by all flows, base-RTT^i is defined as the RTT of flow i in an empty system, and RTT_t^i and rate_t^i are respectively the RTT and transmission rate of flow i at time t. $\frac{\text{RTT}_t^i}{\text{base-RTT}^i}$ is also called the RTT inflation of agent i at time t. The ideal reward is obtained when $\textbf{target} = \frac{\text{RTT}_t^i}{\text{base-RTT}^i} \cdot \sqrt{\text{rate}_t^i}$. Hence, when the **target** is larger, the ideal operation point is obtained when $\frac{\text{RTT}_t^i}{\text{base-RTT}^i} \cdot \sqrt{\text{rate}_t^i}$ is larger. As increasing the transmission rate increases network utilization and thus the observed RTT, the two grow together. Such an operation point is less latency sensitive (RTT grows) but enjoys better utilization (higher rate). As Proposition 1 shows, maximizing this reward results in a fair solution.

Proposition 1. *The fixed-point rate (solution) for all N flows sharing a congested path is* $\frac{\textit{max rate}}{N}$.

HPC for big AI: where do (whitebox) performance models fit?

ToAAS 2023:

Model-driven Cluster Resource Management for AI Workloads in Edge Clouds

QIANLIN LIANG, WALID A. HANAFY, AHMED ALI-ELDIN, and PRASHANT SHENOY,

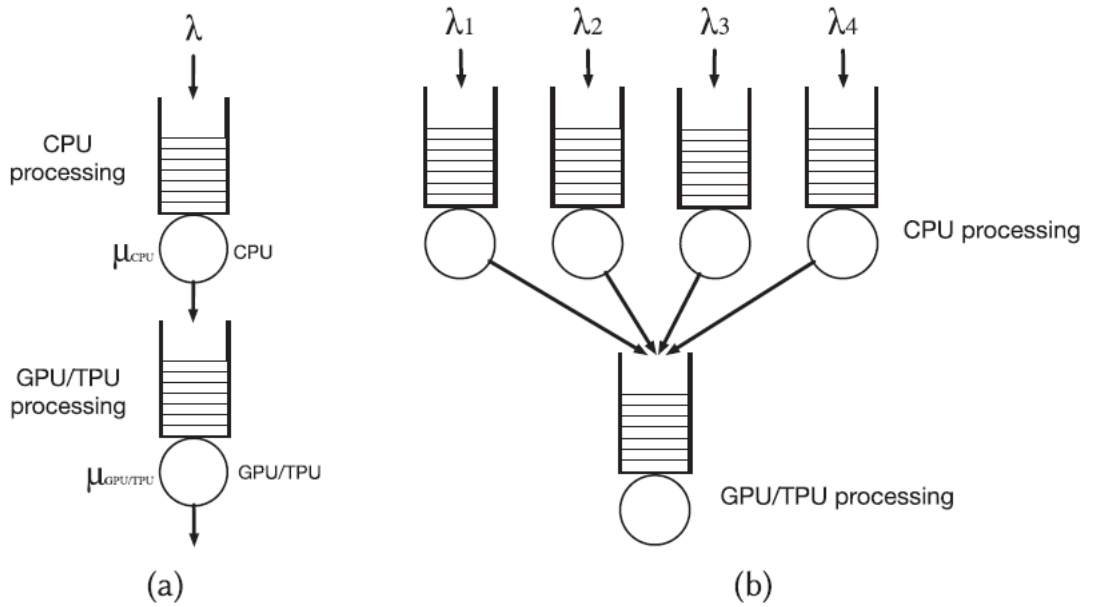


Fig. 2. (a) Tandem queue model for CPU and GPU/TPU processing of a request. (b) Network of queues model showing one CPU queue per application and a single GPU/TPU queue for all applications.

HPC for big AI: where do (whitebox) performance models fit?

EuroSys 2019:

GRNN: Low-Latency and Scalable RNN Inference on GPUs

Connor Holmes, Daniel Mawhirter Yuxiong He

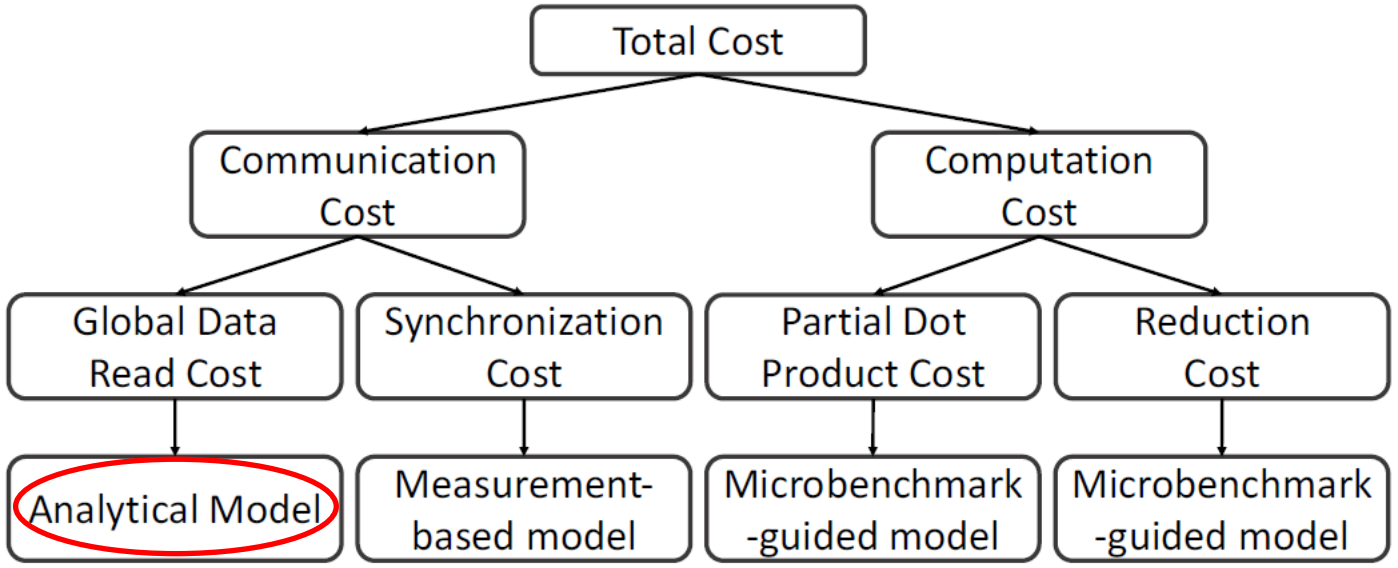


Figure 9. Overview of the performance model.

HPC for big AI: where do (whitebox) performance models fit?

HPCA 2023:

MPress: Democratizing Billion-Scale Model Training on Multi-GPU Servers via Memory-Saving Inter-Operator Parallelism

Quan Zhou[1], Haiquan Wang[1], Xiaoyan Yu[1], Cheng Li[1,2], Youhui Bai[1], Feng Yan[3], Yinlong Xu[1,2]

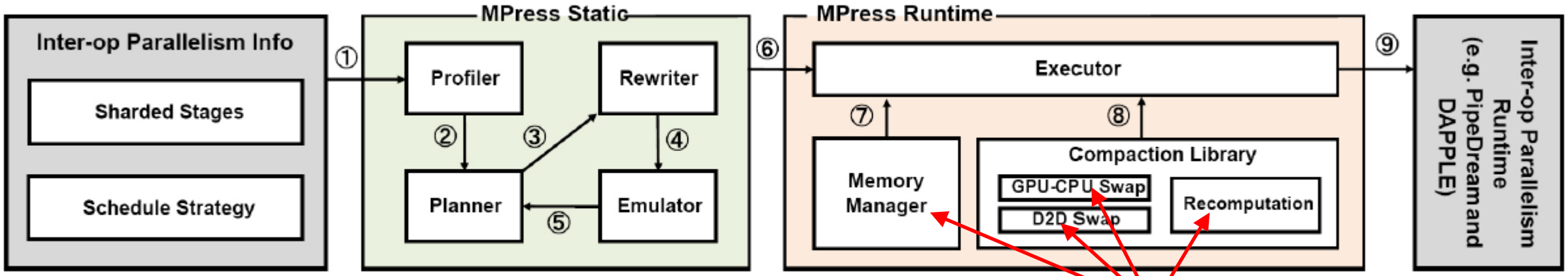


Fig. 5: MPress training system overview.

Reflections and Perspectives on Avenues to Performance

(Reflection) What have the main novelties been during the past decade, in all aspects of HPC?

(Perspective) What is our educated guess on the key developments for the next decade?

- Can whitebox performance models help in scaling performance for ML?
- Should undergrads and PhD students, engineers and researchers learn whitebox modeling?
- What/How should they learn?

TeaPACS 2021

International Workshop on Teaching Performance Analysis of Computer Systems

Milano, Italy • November 12, 2021

https://teapacs.github.io/2021/

Invited Talks:

1. **Mor Harchol-Balter (CMU)**
   The most common queueing theory questions asked by computer systems practitioners
2. **Chee Wei Tan (City Univ. Hong Kong)**
   The value of cooperation: from AIMD to flipped classroom teaching
3. **Cathy Xia (Ohio State Univ.)**
   Teaching performance modeling via software and instructional technology
4. **Jean-Yves Le Boudec (EPFL)**
   Performance evaluation as preparation for statistics and data science
5. **Giuseppe Serazzi (Politecnico di Milano)**
   Updating the content of performance analysis textbooks

Discussions:

1. **The current situation**
2. **What we can do**

TeaPACS 2023

International Workshop on Teaching Performance Analysis of Computer Systems

Orlando, FL, USA • June 19, 2023

https://teapacs.github.io/2023/

Invited Talks:

1. **Giuliano Casale (Imperial College London)**
   Performance evaluation teaching in the age of cloud computing
2. **Diwakar Krishnamurthy (Univ. of Calgary)**
   Teaching software performance evaluation to undergrads: lessons learned and challenges
3. **Mohammad Hajiesmaili (Univ. Massachusetts, Amherst)**
   Teaching learning-augmented algorithms with societal design criteria
4. **Ziv Scully (Cornell Univ.)**
   The role of advanced math in teaching performance modeling

Discussions:

1. **What to teach**
2. **How to teach**

Reflections and Perspectives on Avenues to Performance

(Reflection) What have the main novelties been during the past decade, in all aspects of HPC?

(Perspective) What is our educated guess on the key developments for the next decade?

- Can whitebox performance models help in scaling performance for ML?
- Should undergrads and PhD students, engineers and researchers learn whitebox modeling?
- What/How/Where should they learn?

ScalPerf'23

Between ML and HPC: Where do (whitebox) performance models fit?

Y.C. Tay
National University of Singapore