5th PERMAVOST Workshop
July 20, 2025
Notre Dame, IN

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Towards a Workflow for Analytic Performance, Power, and Energy Models

Georg Hager

Erlangen National High Performance Computing Center (NHR@FAU)

Erlangen National HPC Center (NHR@FAU)

Erlangen National High Performance Computing Center
https://nhr.fau.de

- HPC systems and Infrastructure
- User support
- Training & Teaching
- Research

Funding (2021 – 2030): Approx. € 70 M

NHR Alliance

- 9 HPC centers at German universities
- NHR Graduate School

https://www.nhr-verein.de/en

NATIONALES HOCHLEISTUNGS RECHNEN

Agenda

- Performance modeling approaches
- The potential for automation in resource modeling
- What is missing

Performance models and insights

Purely predictive analytic model

- Direct insight into bottlenecks from first principles
- Model failures challenge model assumptions or input data
- Refinements lead to better insights

Phenomenological analytic model

- Insight with some “uncharted territory”
- Model failure points to inaccurate or unsuitable measurements

Curve-fitting analytic model

- Yields predictions
- Model failure indicates shortcomings of fitting approach
- Refinements by using more fit parameters

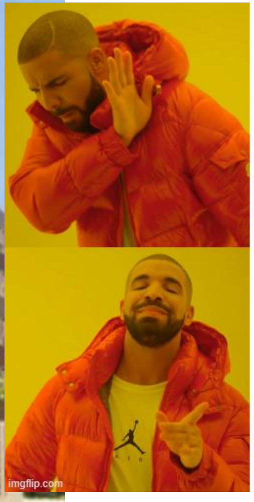
Analytical, Resource-Based, First-Principles Performance Model?

a.k.a. white-box models

A mathematical representation of hardware-software interaction based on simplified machine and application models, which predicts the performance or runtime of a program using hardware resource limits and code requirements

White box: moving a computing center

Source: Akademie Aktuell 02/2006



Copying data over the network

Using trucks filled with tapes/disks

Source: LRZ Garching

$$T_{transfer} = \lambda + \frac{V}{B}$$

Do the math.

Examples for white-/gray-box models in computing

program speedup

serial fraction

$$S(N) = \frac{1}{s + \frac{1-s}{N} + c(N)}$$

Amdahl’s Law with communication

latency

msg. length

bandwidth

$$T_{PtP} = T_l + \frac{L}{B}$$

Hockney model for message transmission time

time for computation

time for data transfer

$$T_{\mathrm{exec}} = \max(T_{\mathrm{calc}}, T_{\mathrm{data}})$$

Roofline model for loop code execution time

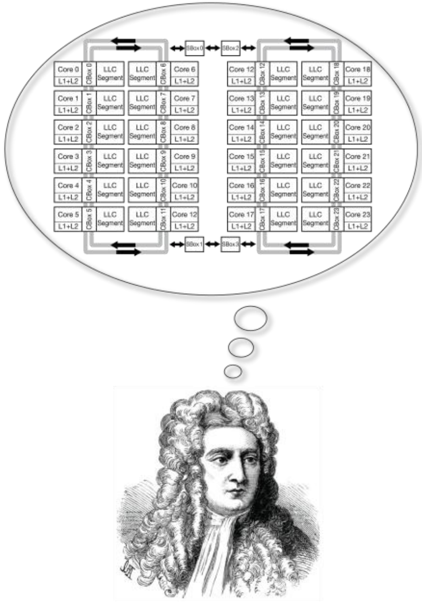
non-overlapping execution

time for data transfer

$$T_{\mathrm{exec}} = f(T_{nOL}, T_{\mathrm{data}}, T_{OL})$$

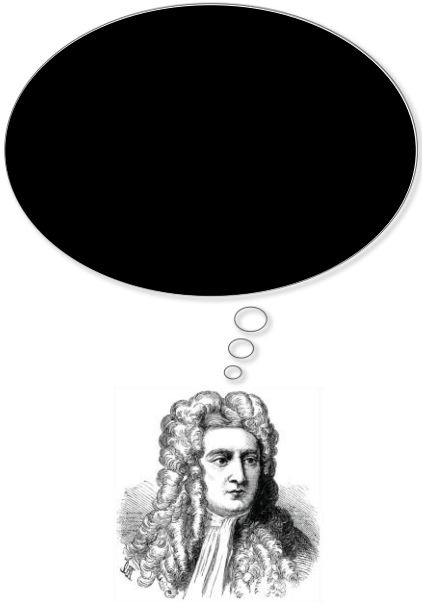
overlapping execution

ECM model for loop code execution time



Motivation for black-box analytic modeling

- White-box models are based on strict assumptions, e.g.:
  - Full overlap of execution & data transfer
  - Steady-state, i.e., ignore wind-up effects
  - Hardware simplifications
- Black-box models have much fewer restrictions
  - Anything that works is allowed
  - Still some assumptions possible



- Black-box performance models
  - Determine influencing factors
  - Deliver target metric predictions for analysis of inaccessible parameter intervals

Performance model normal form

TECHNISCHE UNIVERSITÄT DARMSTADT

$$f(p) = \sum_{k=1}^{n} c_k \cdot p^{i_k} \cdot \log_2^{j_k}(p)$$

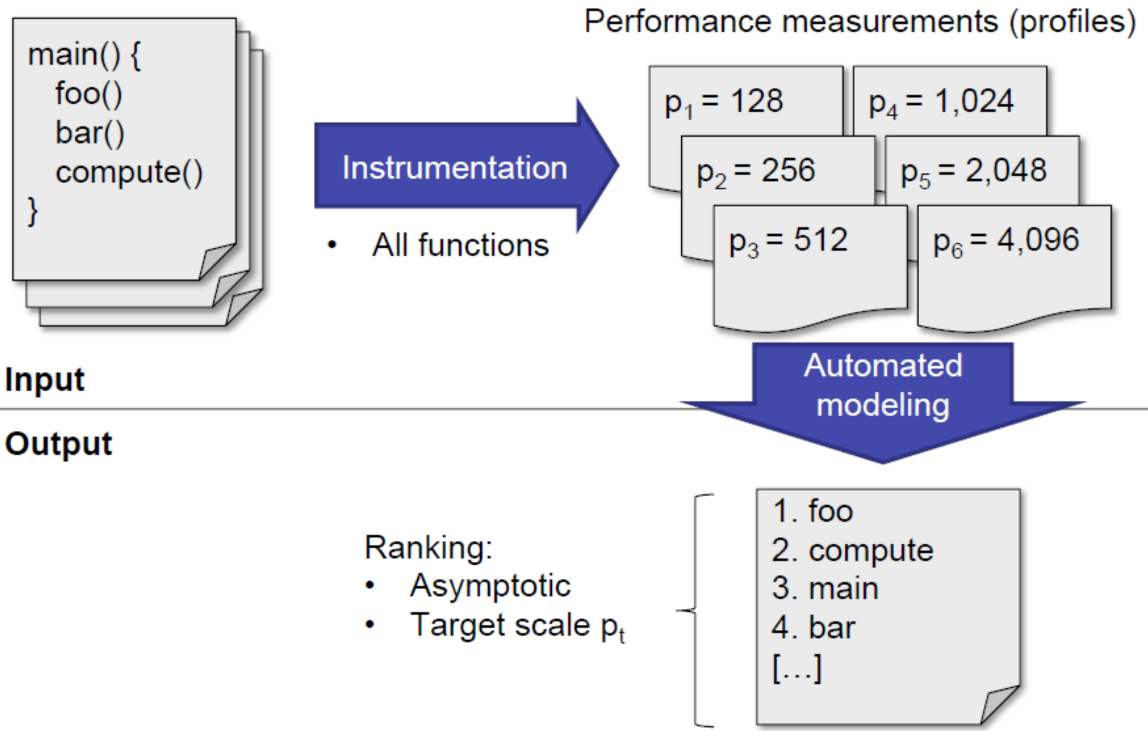
$n \in \mathbb{N}$
$i_k \in I$
$j_k \in J$
$I, J \subset \mathbb{Q}$

$n = 1$
$I = \{0,1,2\}$
$J = \{0,1\}$

c_1 | $c_1 \cdot \log(p)$
$c_1 \cdot p$ | $c_1 \cdot p \cdot \log(p)$
$c_1 \cdot p^2$ | $c_1 \cdot p^2 \cdot \log(p)$

Extra-p
https://github.com/extra-p/extrap

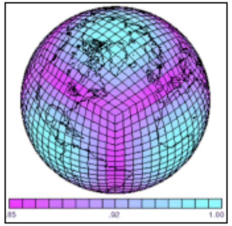
Automated empirical modeling (2)



HOMME – Climate

Core of the Community Atmospheric Model (CAM)

- Spectral element dynamical core on a cubed sphere grid

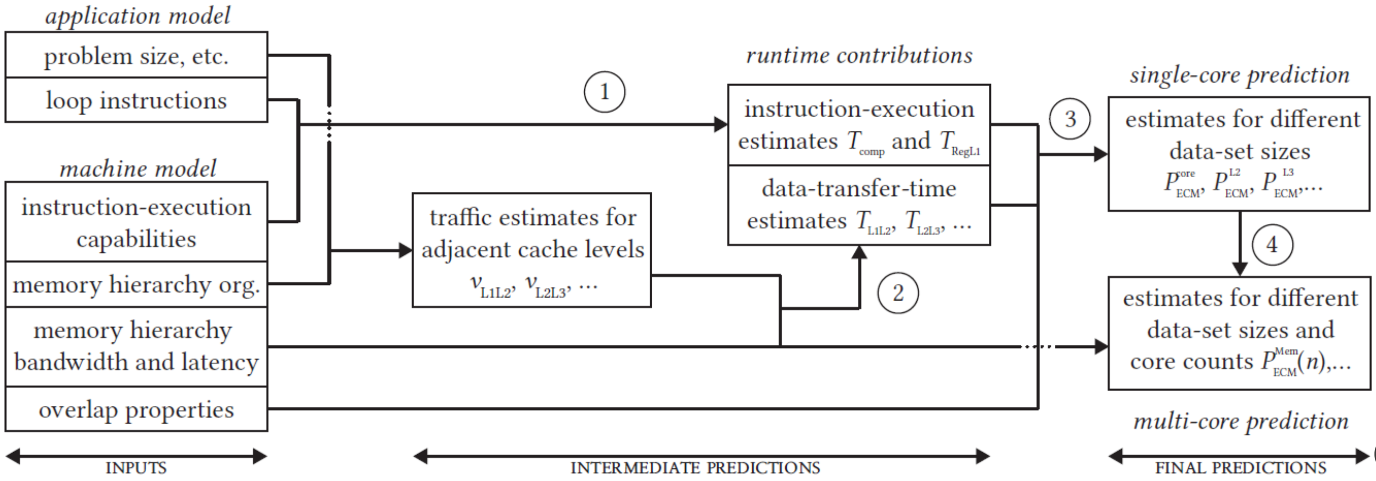


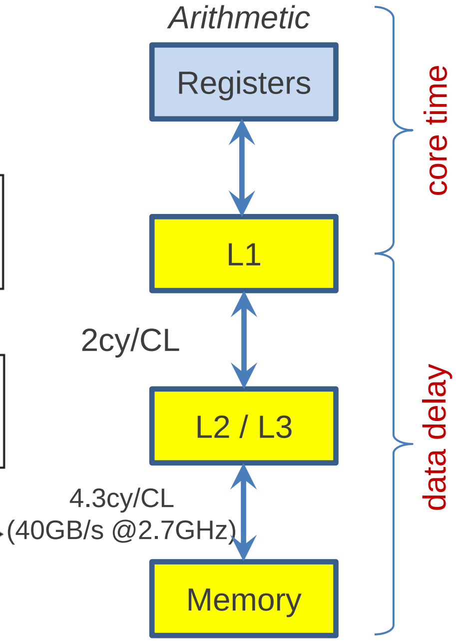
Kernel [3 of 194]	Model [s] t = f(p)	Predictive error [%] p_t = 130k
box_rearrange → MPI_Reduce	$3.63 \cdot 10^{-6} p \cdot \sqrt{p} + 7.21 \cdot 10^{-13} p^3$	30.34
vlaplace_sphere_vk	$24.44 + 2.26 \cdot 10^{-7} p^2$	4.28
compute_and_apply_rhs	49.09	0.83

$p_i \le 43\text{k}$

Example: ECM modeling workflow for loops

J. Hofmann, C. L. Alappat, G. H., D. Fey, G. Wellein, DOI: 10.14529/jsfi200204

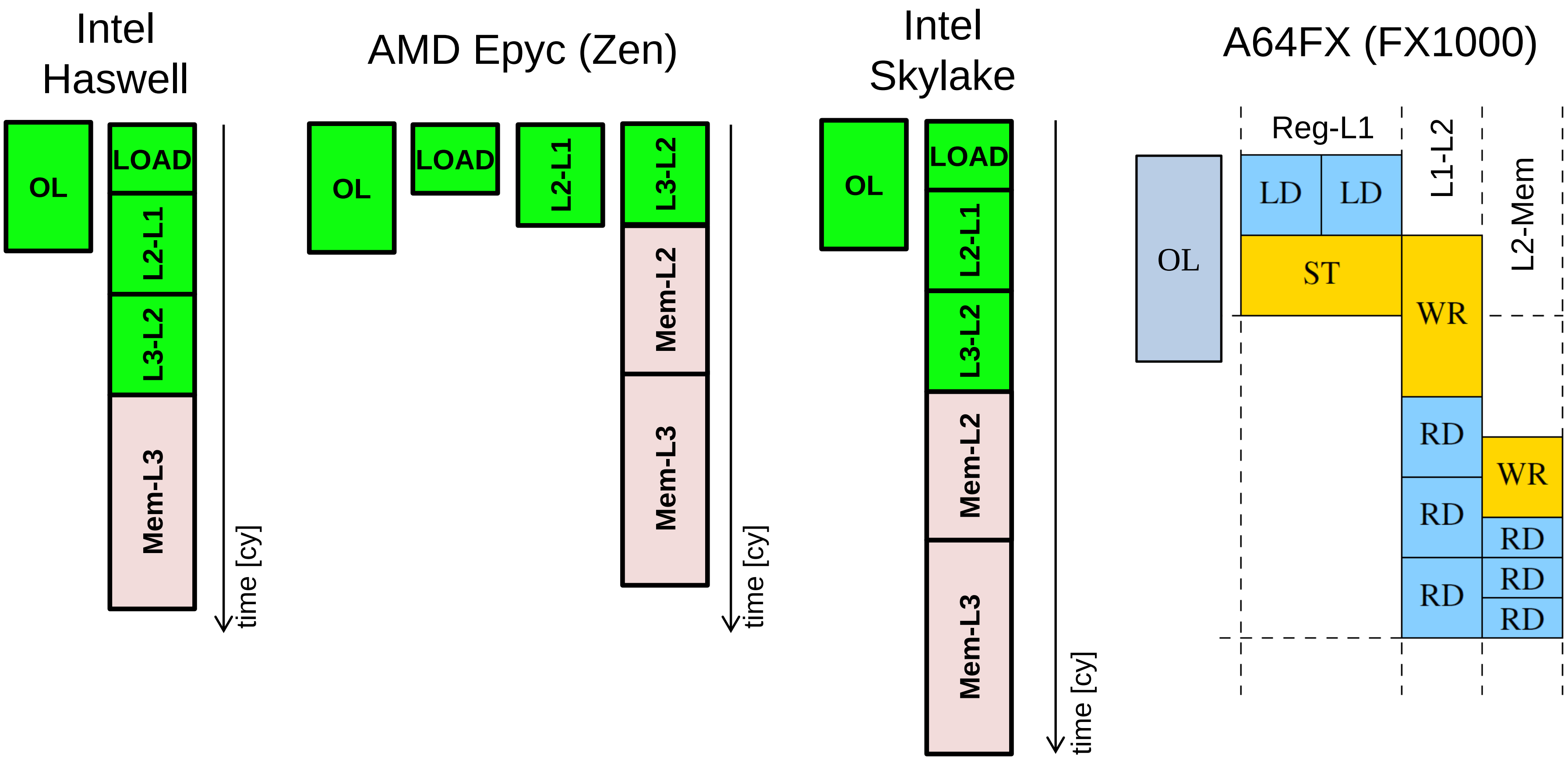




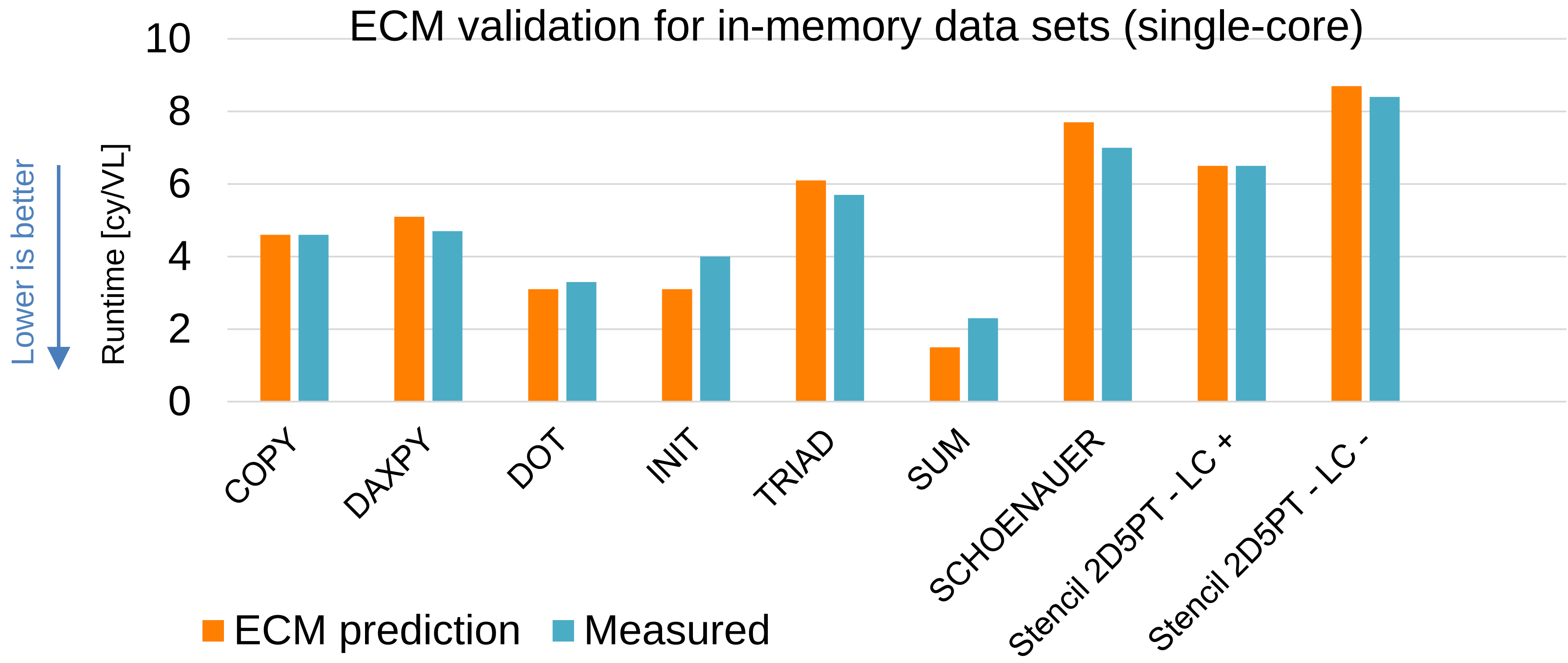
Automating this workflow is possible in some cases:

J. Hammer, J. Eitzinger, G. H., G. Wellein, DOI: 10.1007/978-3-319-56702-0_1 (Kerncraft)
J. Laukemann, J. Hammer, G. H., G. Wellein, DOI: 10.1109/PMBS49563.2019.00006 (OSACA)

Overlap assumptions



Model validation (FX1000, large pages)

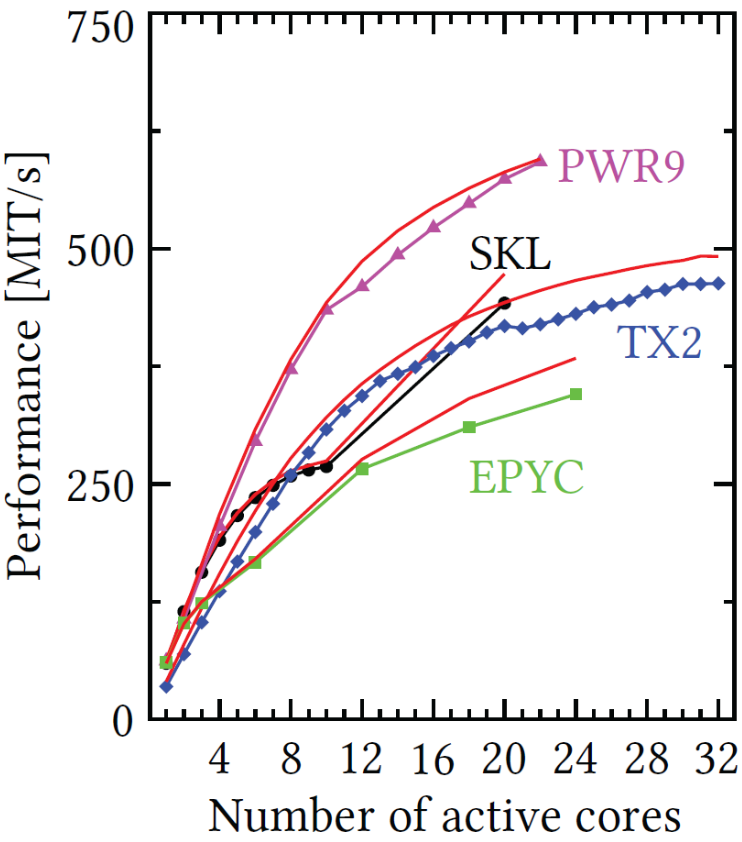


Does it work for “real” code, too?

- Preconditioned matrix- free conjugate-gradient solver
- Four systems
  - IBM Power9
  - Cavium/Marvell TX2
  - AMD Naples
  - Intel Skylake

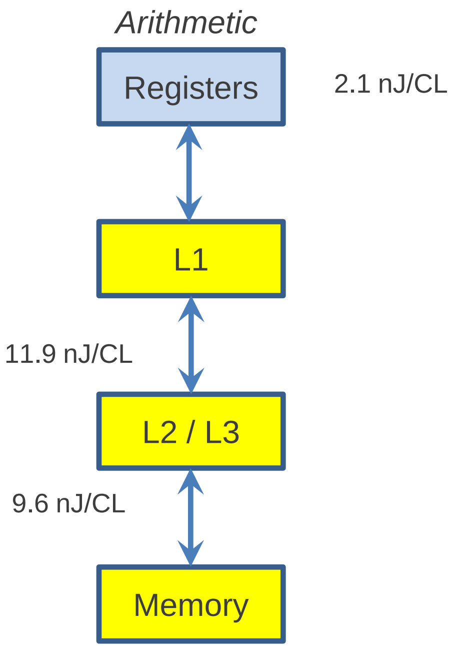
- Yes it does.

J. Hofmann et al., DOI: [10.14529/jsfi200204](https://doi.org/10.14529/jsfi200204)



How about energy modeling? Two approaches!

Based on energy quanta



J.W. Choi et al
DOI: 10.1109/IPDPS.2013.77

Based on power-frequency model

$$P_{\text{base}}(f_{\text{uncore}}) = W_0^{\text{base}} + W_1^{\text{base}} \cdot f_{\text{uncore}} + W_2^{\text{base}} \cdot f_{\text{uncore}}^2$$

$$P_{\text{core}}(f_{\text{core}}) = W_0^{\text{core}} + W_1^{\text{core}} \cdot f_{\text{core}} + W_2^{\text{core}} \cdot f_{\text{core}}^2$$

- Need to determine fit parameters for every loop/code
- $E = P(\{f_i\}) \times T$
  → performance model requred!

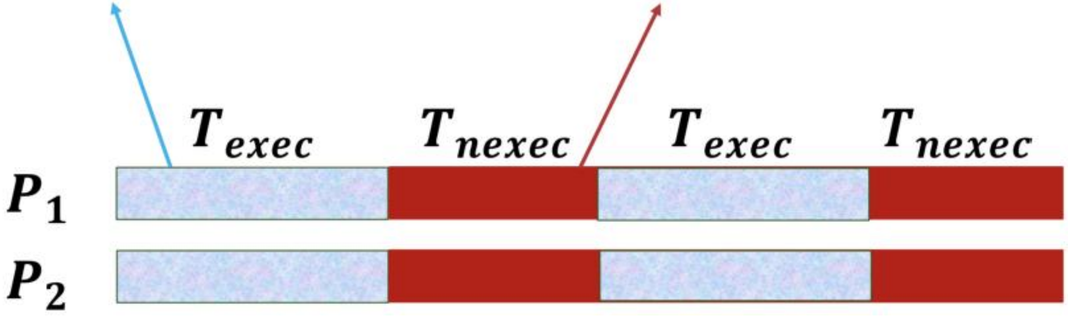
J. Hofmann et al.
DOI: 10.1007/978-3-319-92040-5_2

Beyond the node level: composite analytic models

Plausible assumption: $T = T_{exec} + T_{nexec}$

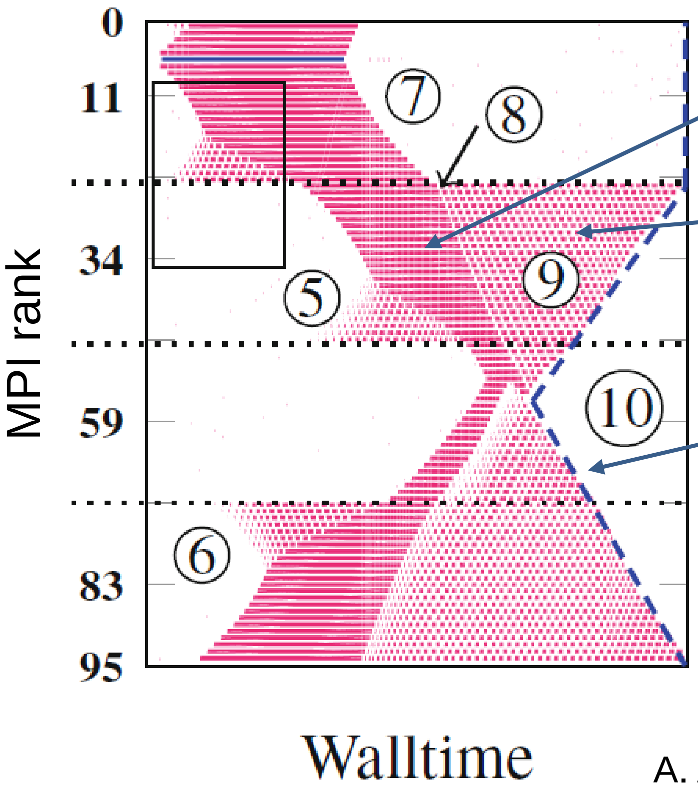
e.g., $\max\left(T_{BW}, T_{flops}\right)$

e.g., $\lambda + \frac{V}{B}$



In practice, $T \neq T_{exec} + T_{nexec}$ and it can go in either direction

Example: computational waves in memory-bound programs



- Decaying idle wave leaves many processes desynchronized
- Inter-process skew → automatic potential communication overlap
- Computational wavefront == rank-time location of all processes at a given iteration
- Memory boundedness is a prerequisite

A. Afzal et al.
DOI: [10.1007/978-3-030-50743-5_20](https://doi.org/10.1007/978-3-030-50743-5_20)

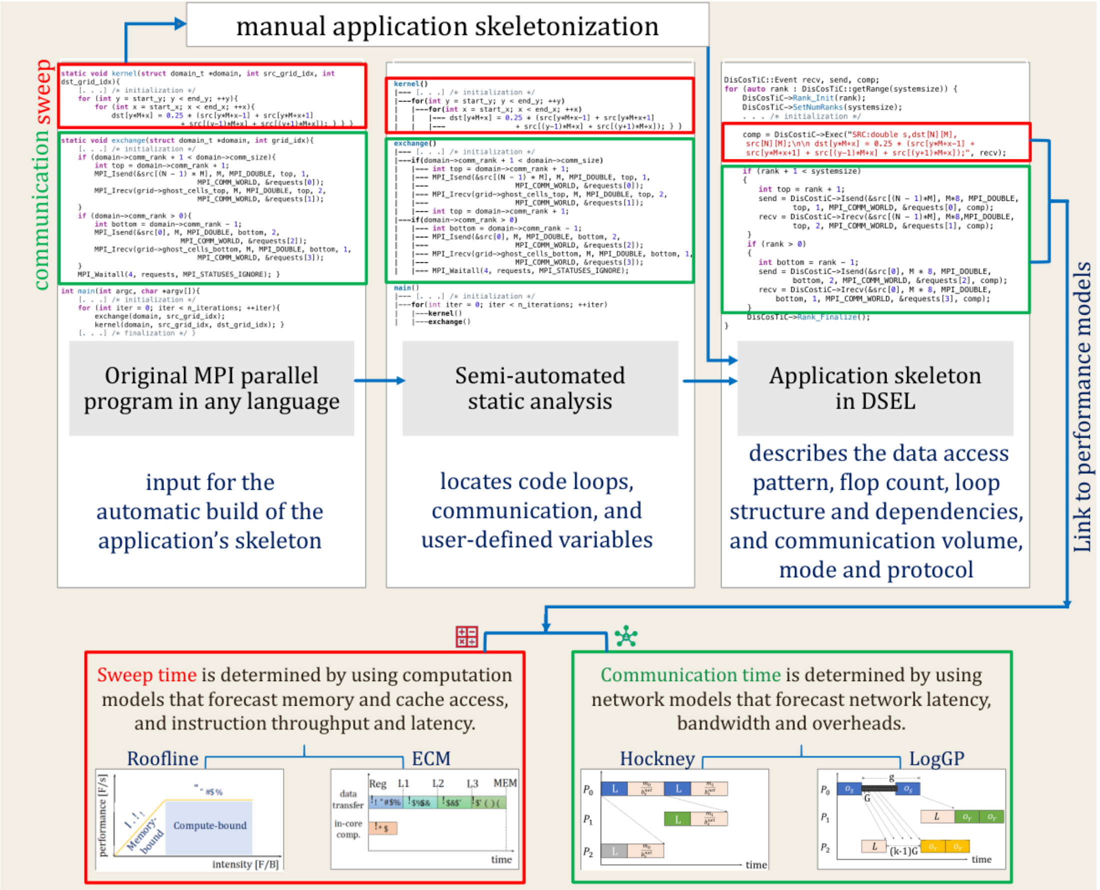
Automated white-box modeling?

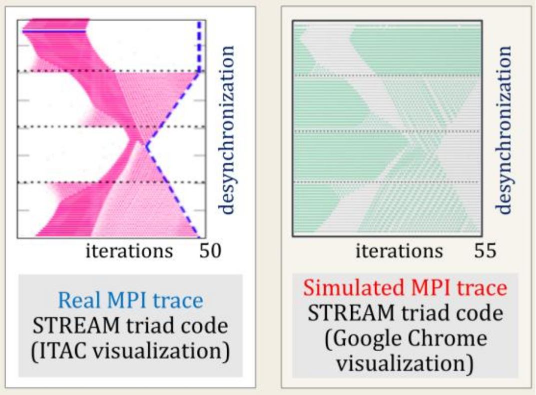
- We need "digital twins" of our parallel applications and clusters!

- (Semi-)automated modeling tools are a prerequisite for this
  - Core-level modeling (code execution):
    OSACA github.com/RRZE-HPC/OSACA
  - Chip-level modeling (Roofline, ECM):
    Kerncraft github.com/RRZE-HPC/Kerncraft
  - Cluster-level modeling (chip level + communication):
    DisCostiC github.com/RRZE-HPC/DisCostiC-Sim

- Still missing: Automated energy modeling beyond curve fitting

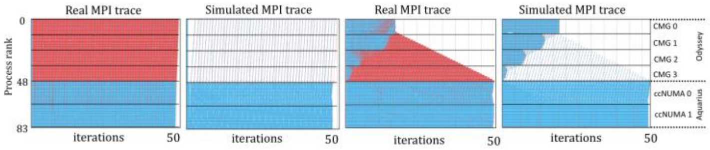
DisCostiC workflow





A. Afzal et al.:
SC24 Best Poster Candidate

A. Afzal et al.:
ISC25 Best Research Poster Award



Conclusion: What is missing

- Accessible and accurate compiler-assisted code execution modeling
- More generic loop nest modeling for Roofline and ECM
- Automatic application skeletonization
- Integrate microbenchmarking for performance and energy
- **We are looking for collaborators!**

Friedrich-Alexander-Universität
Erlangen-Nürnberg

Thank you