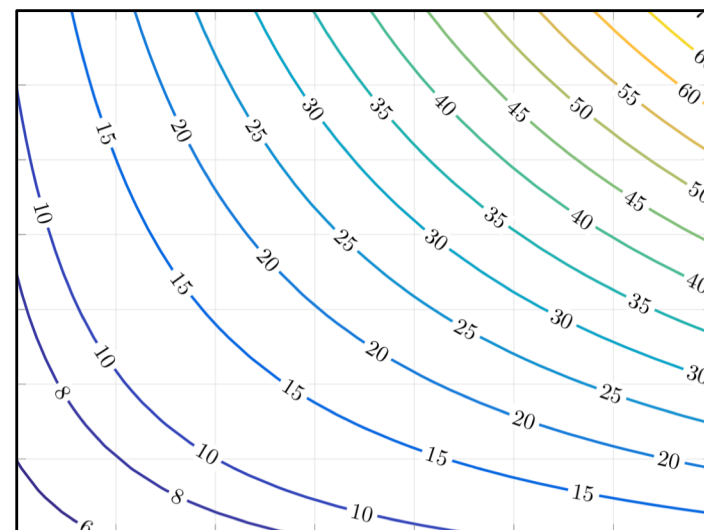
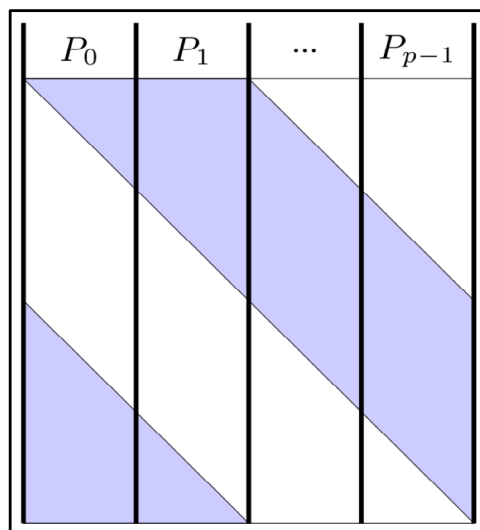
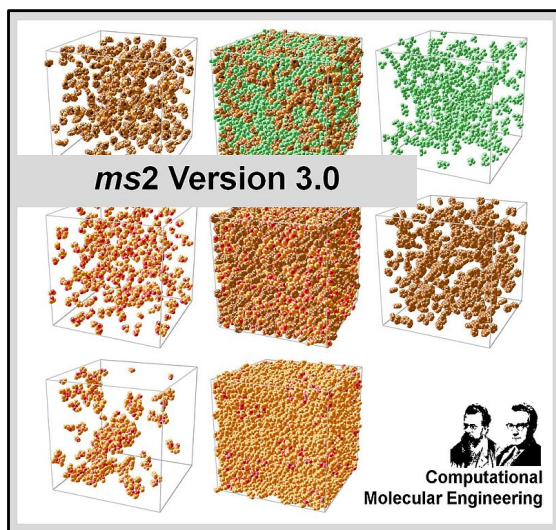


Understanding the Scalability of Molecular Simulation using Empirical Performance Modeling

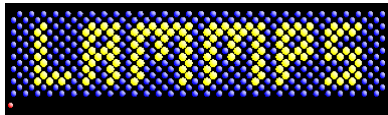
Sergei Shudler¹, Jadran Vrabec², Felix Wolf³



¹ TU Darmstadt (now at Argonne Natl. Lab.), ² TU Berlin, ³ TU Darmstadt

Molecular Dynamics (MD)

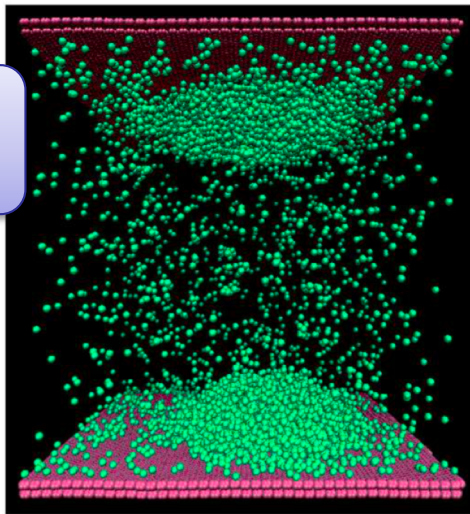
- MD is a method for studying the behavior of matter on the molecular level (biology, materials, biochemistry, ...)
- Well-established simulation packages
 - LAMMPS: Advance research in molecular simulation
 - NAMD: Large biomolecular systems (e.g., viruses)
 - GROMACS: Protein folding



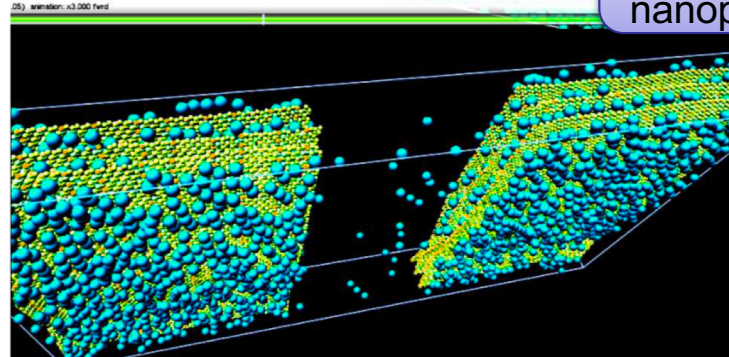
Computational Molecular Engineering (CME)

- Sub-field of MD and simulation-based engineering
- Focuses on the needs of industrial users in chemical and process engineering
- Emergence of sufficiently accurate molecular models of fluids and solids spurred CME

Sessile argon droplet on a solid substrate



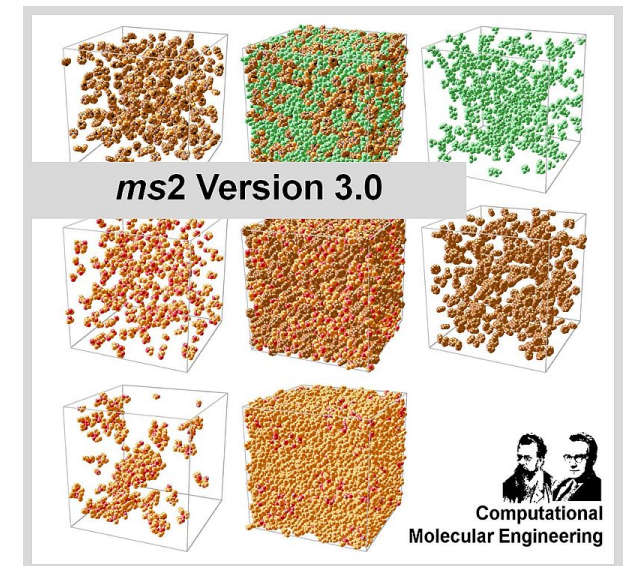
Permeability of fluid methane in nanoporous carbon



Horsch et al. "Computational Molecular Engineering as an Emerging Technology in Process Engineering" (CoRR '13)

ms2 application

- ms2 - molecular simulation: 2nd generation
- Fortran code developed primarily by:
 - University of Paderborn (now TU Berlin)
 - TU Kaiserslautern
 - HLRS (Supercomputing Center Stuttgart)
- Calculates (samples) the full set of thermodynamic properties of bulk fluids
- Goal: scale the temporal range while keeping the time-to-solution as low as possible



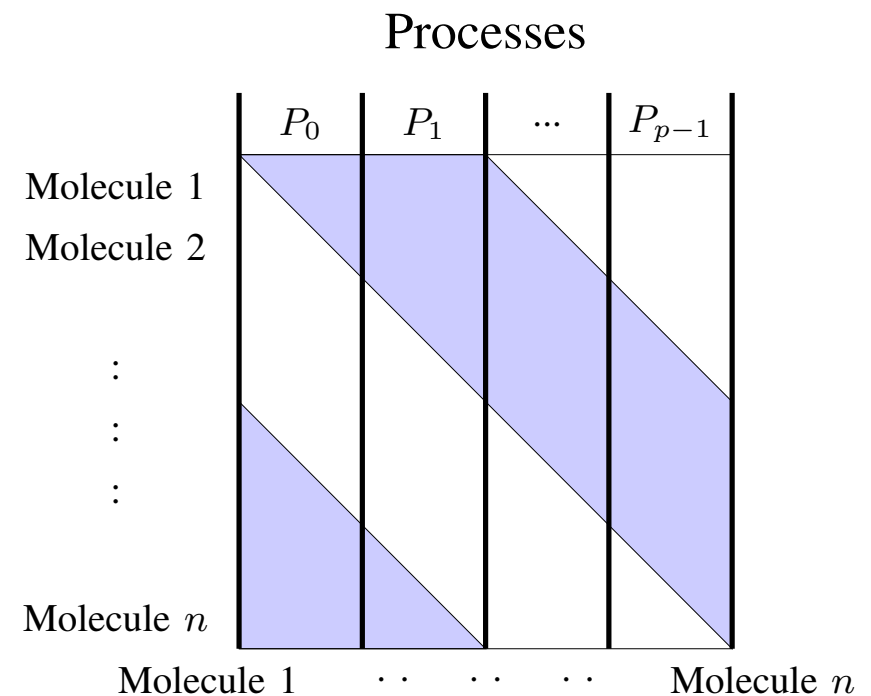
<http://www.ms-2.de>

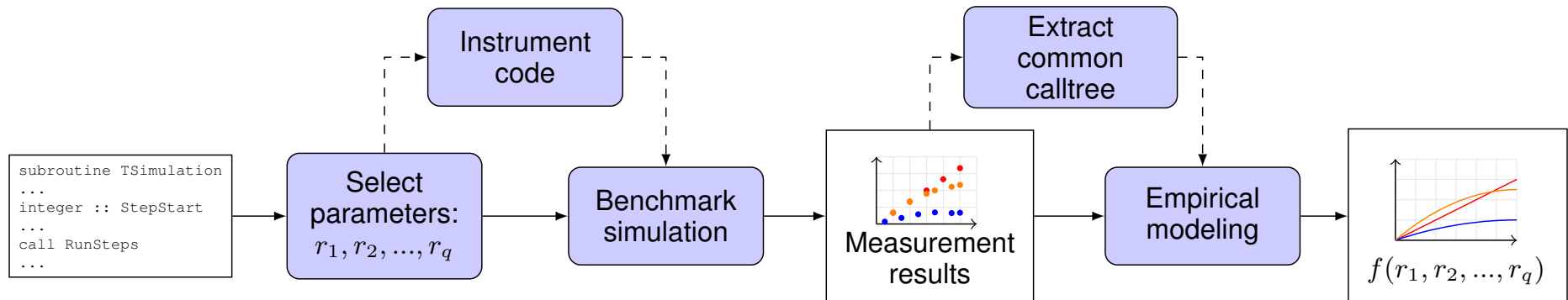
ms2 application (contd.)

- Provides two techniques: Monte-Carlo (MC) and MD
- Each iteration in MC:
 - Molecules are displaced randomly
 - Probability of accepting a displacement is chosen in a way that allows obtaining a representative set of configurations
- Iterations form a Markov chain of configurations
- From this chain a set of static (time-independent) thermodynamic properties are obtained
- Advantage: embarrassingly-parallel problem
- Disadvantage: only static properties

ms2 application (contd.)

- Provides two techniques: Monte-Carlo (MC) and MD
- MD parallelization relies on force decomposition
 - Rearrange interaction matrix
 - Interacting molecules are almost equally distributed
 - Each process has a local copy of molecule data (coordinates, momenta, etc.)
 - Reduction used to sum up all forces exerted on each molecule
- MPI used for both MC and MD



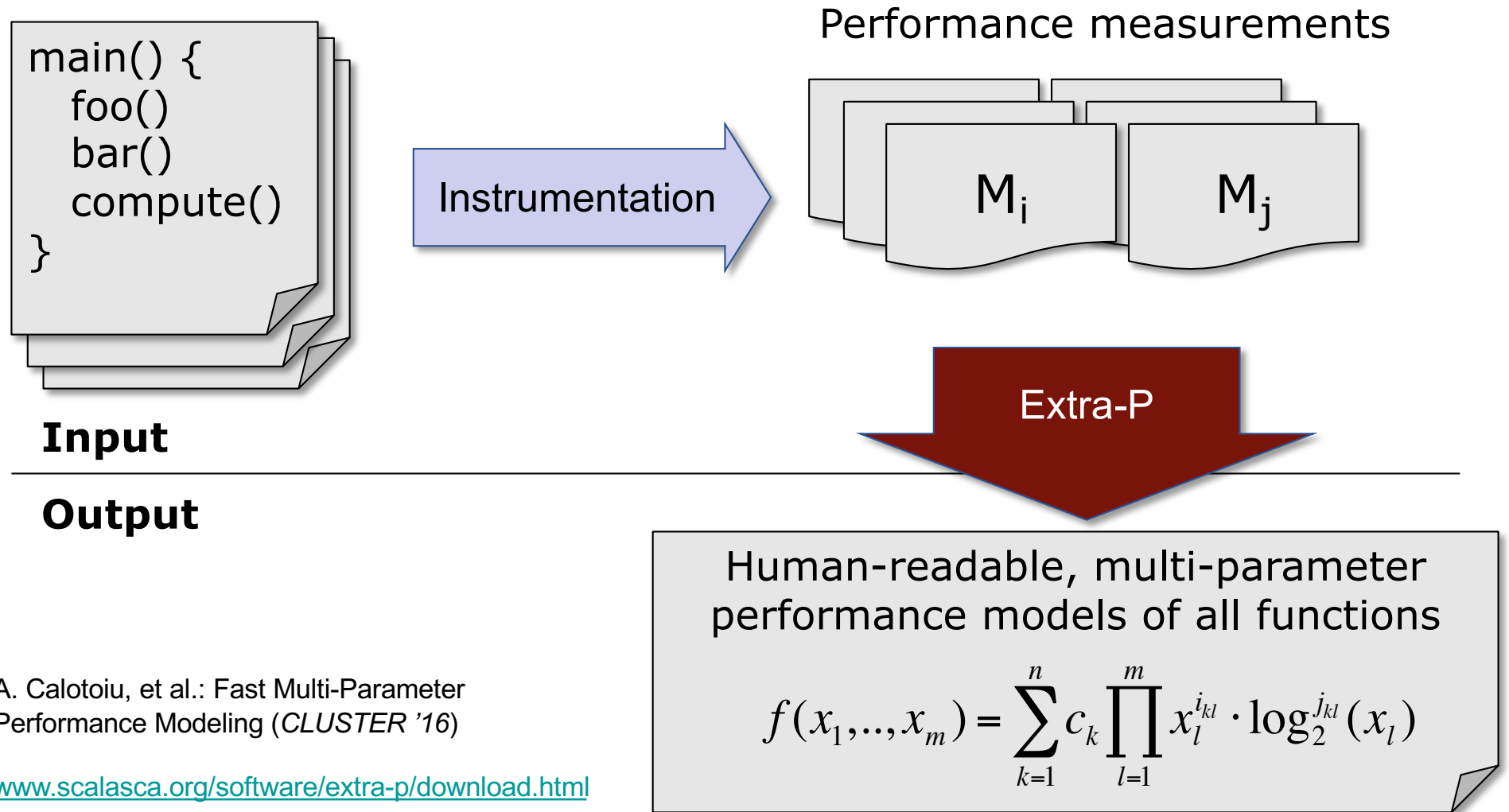


- Goal: produce empirical performance models for the execution time of `ms2`
- Three main phases:
 - Selecting parameters (i.e., independent variables)
 - Benchmarking
 - Empirical modeling

Empirical performance modeling with Extra-P



TECHNISCHE
UNIVERSITÄT
DARMSTADT



A. Calotoiu, et al.: Fast Multi-Parameter Performance Modeling (CLUSTER '16)

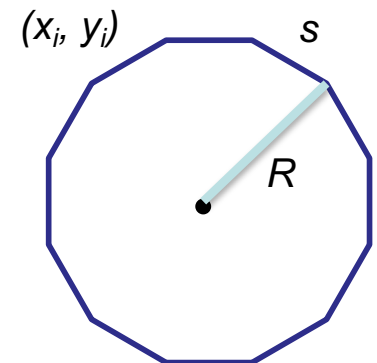
www.scalasca.org/software/extra-p/download.html

Simulation parameters

- Molecular model describes a molecule species:
 - Number and placement of interaction sites
 - Type of interaction site (Lennard-Jones, point charge, ...)
- Identified a group of independent parameters:
 - ***n*** – number of molecules
 - ***m*** – number of interaction sites
 - ***d*** – density of the fluid
 - ***c*** – cut-off radius
 - ***p*** – number of MPI processes

Synthetic Lennard-Jones fluid

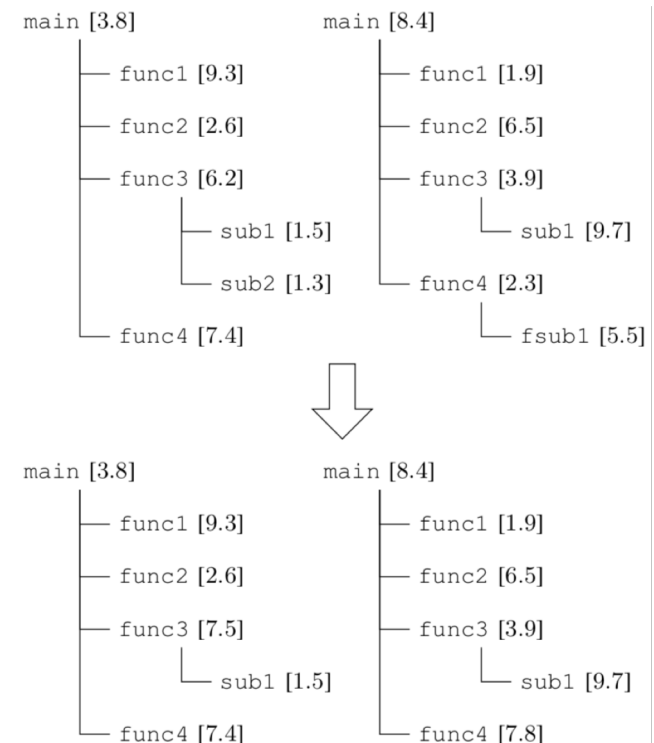
$$R = \frac{s}{2 \sin(\frac{\pi}{m})}$$
$$x_i = R \cos(2\pi \frac{i}{m})$$
$$y_i = R \sin(2\pi \frac{i}{m})$$



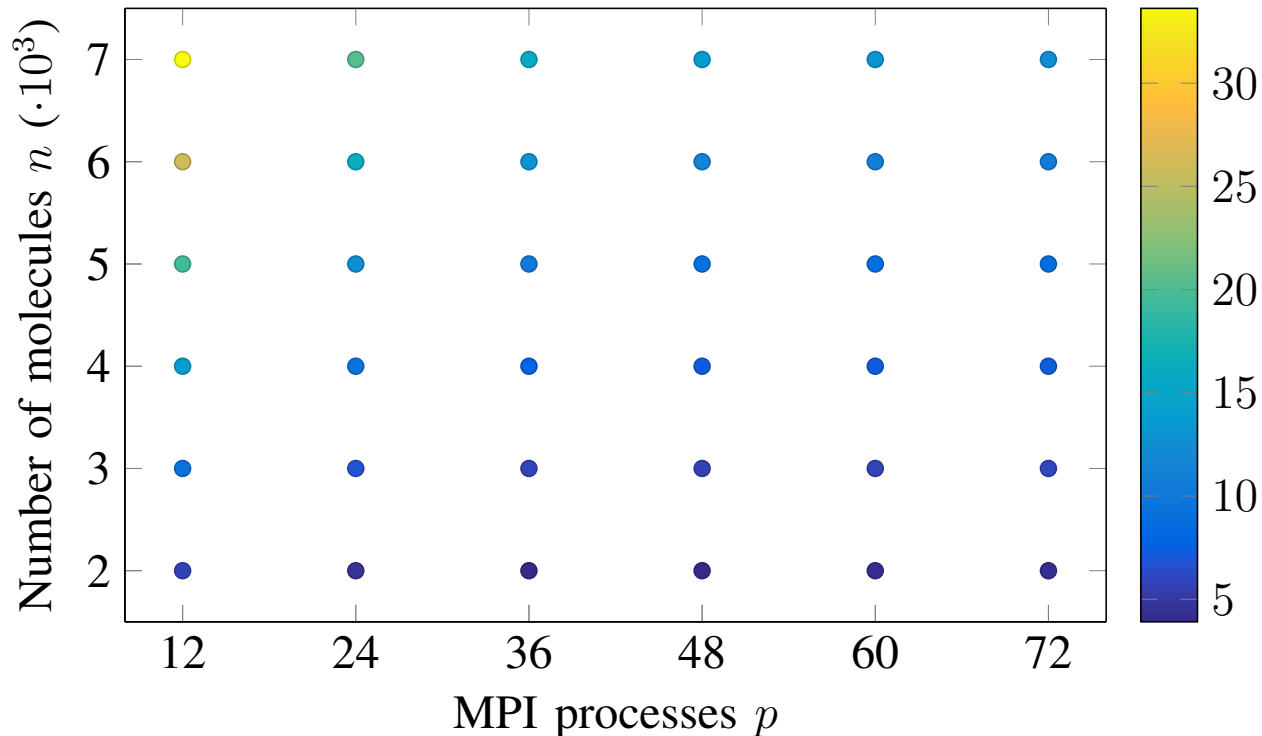
Benchmarking

- Optional: full instrumentation with Score-P*
 - Produces models for every function, kernel, block
 - Requires common calltree extraction
- Run simulations for different values of independent parameters
 - Only 2 or 3 parameters at a time
 - Rule of thumb: at least 5 values
- All other parameters are kept constant

* Score-P: https://www.vi-hps.org/front_content.php?idcat=84



Typical benchmarking result



- Two parameters were varied (p and n)
- Color of each point – execution time in seconds

Experimental setup

- Hazel Hen – Cray XC40 at HLRS
- Node:
 - 2 Intel Xeon E5-2680 v3 processors with 12 cores (24 in total)
 - 128 GB memory – more than 5GB per core
- Aries interconnect
- Dragonfly topology



Measurement variability (Cray XC40)

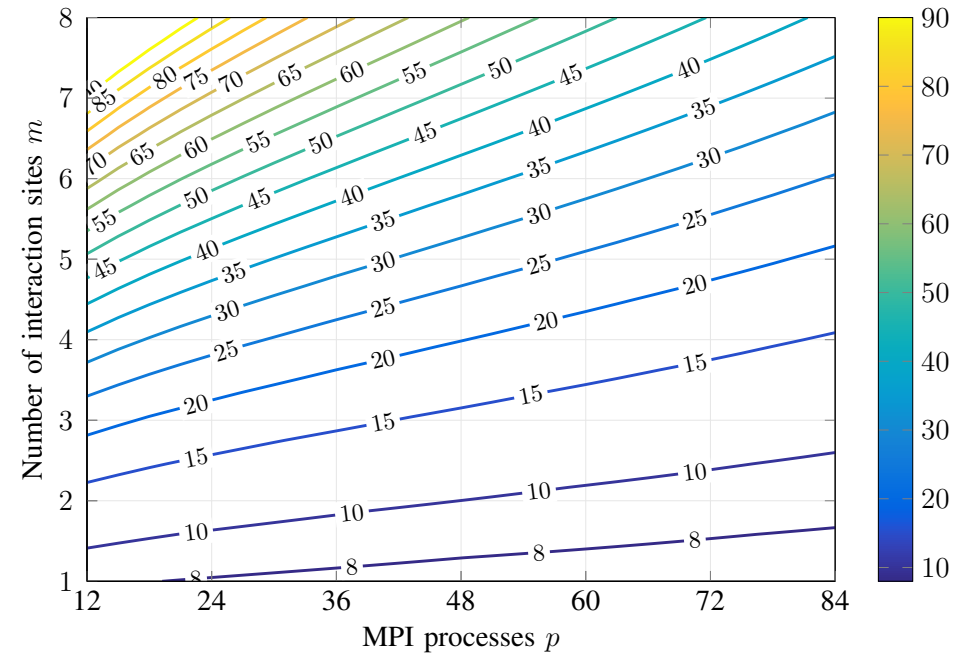
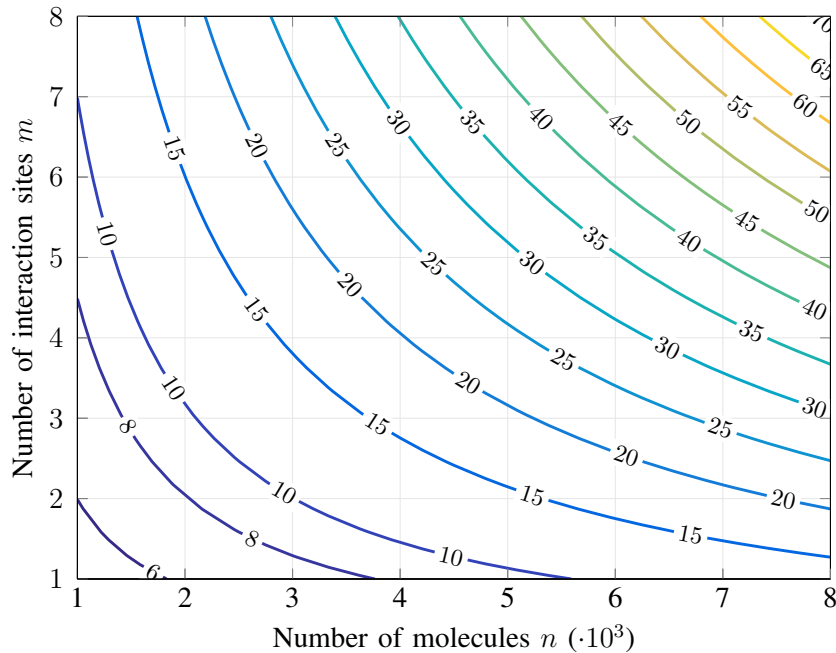
- Dragonfly topology
 - High-radix, low-diameter network
 - Uses adaptive routing and random node placement
 - Alleviates congestion and achieves better load-balancing
- But: applications interfere with each other
- Measurements had between 3.3% to 58.7% CV
(coefficient of variance = $\text{std-dev} / \text{mean}$)
- Factors:
 - Time steps
 - Full instrumentation
 - Compact placement

Chunduri et al. “Run-to-run Variability on Xeon Phi Based Cray XC Systems” (SC ‘17)

Bhatele et al. “Analyzing Network Health and Congestion in Dragonfly-based Supercomputers” (IPDPS ‘16)

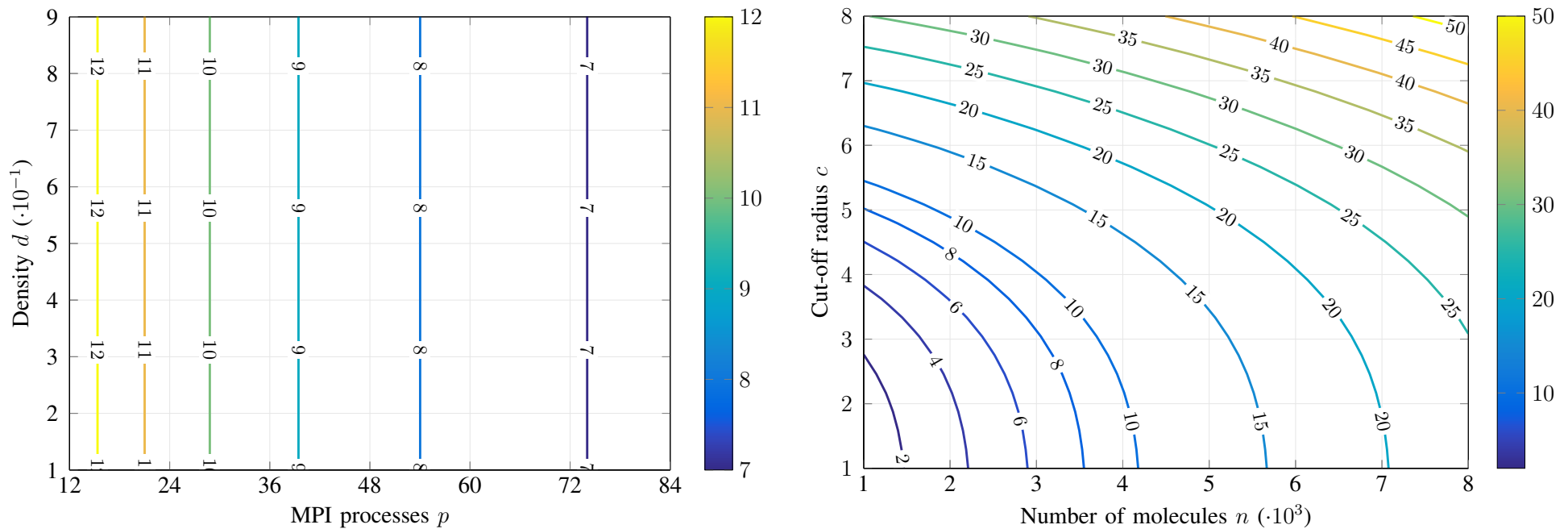
Yang et al. “Watch Out for the Bully!: Job Interference Study on Dragonfly Network” (SC ‘16)

Results (excerpt)



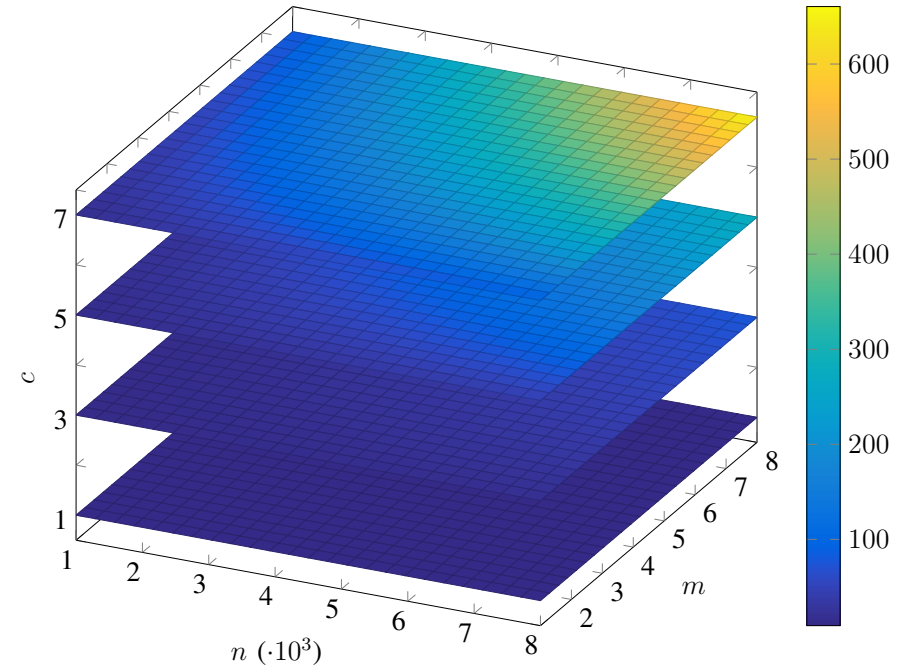
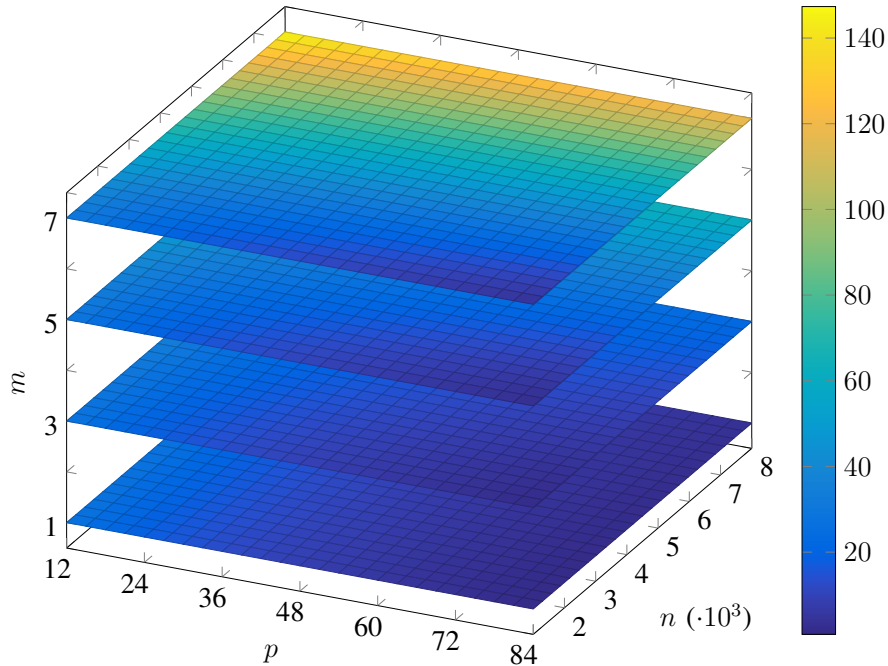
Model params	Fixed params	Model	Adj. R ²
$T(n, m)$	$d, c, p = 72$	$4.41 + 8.03 \cdot 10^{-5} \cdot m \cdot n \cdot \log n$	0.99
$T(p, m)$	$n = 4K, d, c$	$6.6 + 3.21 \cdot m^2 - 0.42 \cdot m^2 \cdot \log p$	0.92

Results (excerpt)



Model params	Fixed params	Model	Adj. R ²
$T(p, d)$	$n = 4K, m = 1, c$	$20.67 - 2.2 \cdot \log p$	0.88
$T(n, c)$	$m = 1, d, p = 36$	$0.06 \cdot c^3 + 1.81 \cdot 10^{-5} \cdot \log^2 n$	0.95

Results (excerpt)



Params	Fixed	Model	Adj. R ²
$T(p, n, m)$	d, c	$62.28 + 2.03 \cdot 10^{-8} \cdot m^2 \cdot n^{1.5} \cdot \log^2 n - 9.63 \cdot \log p$	0.83
$T(n, m, c)$	$d, p = 72$	$9.24 + 5.71 \cdot 10^{-6} \cdot n \cdot \log n \cdot c^2 \cdot \log c \cdot m$	0.88

Discussion

- Models with more than 3 parameters
 - Training set size $5^{\text{no. parameters}}$
 - Sample (reduce) the training set?
- Optimal number of terms for 2 and 3 parameter models
 - More terms leads to better adj. R^2
 - Overfitting (too many features, model too specialized)
- Use machine learning?

Summary and conclusion

- Insight on the impact of parameters on the runtime
- Models can support finding optimal execution configurations
- Pitfalls to watch for in the modeling workflow

- Work supported by:



GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

References (partial):

A. Calotoiu et al.: Fast Multi-Parameter Performance Modeling (*CLUSTER '16*)

S. Shudler et al.: Isoefficiency in Practice: Configuring and Understanding the Performance of Task-based Applications (*PPoPP'15*)

S. Deublein et al.: ms2: A molecular simulation tool for thermodynamic properties (Comp. Phys. Comm. '11)

