Who is afraid of Amdahl’s Law?
Optimizing Scalability by Modular Supercomputing

Thomas Lippert
Director of Centre for Simulation and Data Science
Head of Jülich Supercomputing Centre
Jülich Research Centre, Germany
Professor for Computational Theoretical Physics

Dell-EMC HPC Community Meeting, Austin, Texas

28.3.2017
My Home Base:

JÜLICH RESEARCH CENTRE
Science Campus Jülich

5,700 staff members

- Institutional funding: 320 mio. €
- Third party funding: 238 mio. €

Project management: 1,6 billion €

Teaching:
- ~ 900 Phd students (Campus Jülich)
- ~ 350 Trainees

Research for the future for key technologies of the next generation and Information

28.3.2017

Thomas Lippert
Convergence in the Field „Information“

- Basic research
- Technology
- Translation

Neuroscience

- Neuroscience
- Structural Biology
- Neuronal Information Processing

Machine Learning
- Bioinspired Materials
- Neuromorphic Computing

- Quantum comput.
- Resistive memory
- Spintronics

Information Technologies

- Exascale
- Big Data (Analytics)
- Modeling

High Performance Computing

- Anorganic Components & Architectures
- Energy Efficient Hardware

Materials Simulations

Virtual Design

Brain Simulations

Neural Information Processing

Neural Networks
Convergence of Neuroscience and HPC

Human Brain Project / Decoding the Human Brain

Simulation-Lab
Jülich supercomputers
Visualization
BigData
Neuroinformatics

Local and global receptor distribution

Fiber tracts

Dynamic connectivity

Scientific Director of HBP
Prof. Katrin Amunts, Jülich

BIG DATA ANALYTICS & SIMULATION

Molecules

Electron microscopy
Image analysis
9.4 T animal MR
Tracer development
Radiopharmacology
High-throughput microscopical Imaging

3T MR
9.4 MR-PET
3T MR-PET

NEUROIMAGING

Multiscale and multimodal in space and time

28.3.2017

Thomas Lippert

3.6.2017
JÜLICH SUPERCOMPUTING CENTRE
Mission

Enable scientists and engineers to solve grand challenge problems of highest complexity in science and engineering through information processing by means of Supercomputing and Big Data in collaborative infrastructures.
Challenges

Extreme Scale Computing

Scientific Big Data Analytics

Deep Learning

Education/Training
Activities

- **Operating the basic infrastructure**
  - Supercomputers, software systems, visualization systems, networks, Peta-scale storage

- **Organizing access via peer-reviewed proposals**
  - Coordination office, independent committees, external experts

- **Developing user support**
  - Helpdesk, application support via simulation labs

- **Co-design with users and vendors**
  - Exascale labs, Joint vendor-application labs

- **Integration into international structures**
  - GCS, HBP, PRACE, EUDAT, EGI, ETP4HPC, JLESC
Creating Strategic Infrastructures

- Gauss Centre for Supercomputing
  - Stuttgart – Munich - Jülich

- Partnership for Advanced Computing in Europe
  - PRACE Implementation-Coordinator

- EUDAT
  - Replicate, store and share research data in Europe

- Human Brain Project
  - Leading the HPC Platform
  - Solving the HBP federated data challenges
Support by SimLabs & DataLabs

Communities
- HPC Know-How
- Models
- Fluid & Solid
- Neuroscience
- Plasma
- Terrestrial Systems
- Climate
- Ab Initio
- Molecular Systems
- Biology
- Nuclear & Particle
- Cross-Sectional Teams
- Kernels
- Application Optimisation
- Math. Methods
- Parallel Performance
- Visualisation

Exascale
- Co-Design
- Big Data
- Jülich
- ECL
- NVIDIA

Community-driven Research & Support Hubs

28.3.2017
Thomas Lippert
High-Q-Club: >40 generic codes at scale!

- dynQCD
- Gysela
- JusPIC
- MP2C
- μφ

Non-trivial kernels only!

- PEPC
- PMG+PFASST
- TeraNEO
- WalBerla

Gyrokinetic code
Laser-Plasma
CFD with Particles
Water in Porous Media
Particle Tree Code
ODE-Solver
MG for Geophysics
Lattice Boltzmann etc.
Metrics of Progress:
250 Publications / Year – Major Journals like Science and Nature

D. Marx et al., Nature Chemistry 5 (2013) 685
Bochum

Jülich

Jülich

Sissi de Beer, Nature Communications 5, doi 10.1038/ncomms4781
Twente

Wuppertal

28.3.2017

Thomas Lippert
Past: Supercomputer evolution @ JSC

2004
IBM Power 4+ JUMP, 9 TFlop/s

2009
IBM Power 6
JUMP, 9 TFlop/s
IBM Blue Gene/L
JUBL, 45 TFlop/s
JUROPA
200 TFlop/s
HPC-FF
100 TFlop/s

File Server

2014
IBM Blue Gene/P
JUGENE, 1 PFlop/s
IBM Blue Gene/L
JUBL, 45 TFlop/s
JUQUEEN
5.9 PFlop/s

JURECA
2 PFlop/s
+ Booster
~ 5 PFlop/s

Intel - DELL

2019
IBM Blue Gene/Q
JUQUEEN
successor
Modular System
Highly Scalable

Thomas Lippert
Future: Modular Design Principle

First Step: JURECA will be enhanced by a highly scalable Module
CONCURRENCY TALES
Notation

- $S$: speedup compared to single core
- $N$: # of parallel cores
- $s$: portion computed sequentially
- $p$: portion computed in parallel
- $O(1)$ concurrency = scalar processing
- $O(N)$ concurrency = N parallel processes
Scalar vs. Parallel Code Parts

\[ S_{1,N} = \frac{1}{s + \frac{p}{N}} \]

strong scaling!
\[ s_r = \frac{1}{2} \]

<table>
<thead>
<tr>
<th>Cores (N)</th>
<th>Speedup Factor</th>
<th>Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00 (baseline)</td>
<td>sequential, potentially ( O(N) ) parallelizable, sequential</td>
</tr>
<tr>
<td>2</td>
<td>1.33</td>
<td>sequential, core 1, sequential, core 2</td>
</tr>
<tr>
<td>4</td>
<td>1.60</td>
<td>sequential, core 1, core 2, core 3, core 4</td>
</tr>
<tr>
<td>( \infty )</td>
<td>2.00</td>
<td>sequential, sequential</td>
</tr>
</tbody>
</table>

http://drdobbs.com
Gustafson 1988

- Amdahl keeps workload fixed
  → strong scaling 😞
- But (often) problem size should be larger anyway
- For larger problems, use larger systems
  → weak scaling 😊
- Total work $w$ in fixed time on $N$ cores is
  \[ w_N = s + pN \]
- $(s$ assumed to stay constant with $N$ …)
  \[ S = \frac{w_N}{w_1} \xrightarrow{N\to\infty} \infty \]
<table>
<thead>
<tr>
<th>Cores (N)</th>
<th>Total Work</th>
<th>Core 1</th>
<th>Core 2</th>
<th>Core 3</th>
<th>Core 4</th>
<th>Core 5</th>
<th>Core 6</th>
<th>Core 7</th>
<th>Core 8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0</td>
<td>sequential</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.5</td>
<td>sequential</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.5</td>
<td>sequential</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4.5</td>
<td>sequential</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$s_r = \frac{1}{2}$

http://drdobbs.com

28.3.2017

Thomas Lippert
Scalability

- Quite some applications capable to scale to $O(500k)$ cores
  - E.g. Sparse matrix-vector codes
  - Highly regular communication patterns

- But: Many applications have more complex kernels
  - Complicated communication patterns
  - Well suited for cluster systems

- Reality
  - Mix of code parts with a few different concurrencies
  - → Optimize these concurrencies on the most suitable hardware
Multiple Concurrences

Assume
2 concurrency levels, K and N

\[ S_{K,N} = \frac{1}{1 - p} + \frac{p}{Kf} \]

\( p_r = 0.50, N = 500.000, K = 10.000, f = 1: \) \( \rightarrow S = 20.000 \)

\( p_r = 0.90, N = 500.000, K = 10.000, f = 4: \) \( \rightarrow S = 150.000 \)

\( p_r = 0.99, N = 500.000, K = 10.000, f = 4: \) \( \rightarrow S = 400.000 \)
Many Challenges

- Energy consumption
- Heterogeneity
- Programmability
- Concurrency
- Resiliency
- Exploding data requirements
- Algorithms and application readiness
- etc.

Credo TL:
For a given code, the modular environment adapted to the various concurrency levels is most energy efficient!
ACCELERATION STRATEGIES
Multi Core Processors

- Broad range of capabilities
- Act as general purpose processors
- High single thread performance
- High frequency
- Out of order processing
- High memory per core
- Standard programming env.
- Limited energy efficiency
- Large $ / core
Many Core Processors

Intel Xeon Phi

- O(70) cores, 4 threads / core
- Rather low single thread performance
- In-order architecture, low frequency
- More energy efficient ($/FLOP)
- x86 architecture → standard programming
  Can run autonomously (without host)

GPUs

- Evolve into general purpose
- Hundreds of (weaker) computing cores
- More energy efficient ($/FLOP)
- Specific programming models (OpenCL)
- Require host CPU
Homogeneous cluster

- Cluster Nodes: **general purpose** (multi-core) processor technology
- Same processor characteristics in all nodes
- Single high-speed network connecting all nodes
- Limited scalability due to general purpose ballast
Accelerated cluster

- One or more MCPs attached to each Cluster Node
- Static assignment of accelerators to CPUs
- Accelerators do not act autonomously
- Flat network topology
- Usage of resources always lower 100 %
Cluster of ManyCPs

- Node consists of one or more ManyCPs
- Directly connected to network
- Possible with few accelerator technologies
- Most need CPU to boot and communicate with the network
Modular Concept

- Combine cluster-nodes with autonomous ManyCPs as accelerators
- **Dynamical** assignment of cluster-nodes and accelerators
- Off-load **complex kernels**
  - communication between CPU and Accelerator less demanding
  - larger messages are less sensitive to latency
The DEEP Experiment (2011 - 2015)

DYNAMICAL EXASCALE ENTRY PLATFORM

www.deep-project.eu        www.deep-er.eu        www.deep-est.eu
The DEEP Modular Cluster-Booster-Architecture

Cluster

Low/Medium scalable code parts

Booster

Highly scalable code parts

28.3.2017 Thomas Lippert
Cluster

- Hot-water cooled cluster
- Compute nodes: 128
  - 2 x Intel Xeon (Sandy Bridge) CPU
  - 32 GB RAM
- Networks:
  - InfiniBand (QDR) – main network
  - Fat-tree topology
- Performance:
  - 44 TFlops peak
  - 2.8 GB/s IB bandwidth
Designed and developed from scratch in DEEP

- **Booster Nodes:** 384
  - Intel Xeon Phi (Knights Corner)

- **Network:**
  - EXTOLL (direct non-switched network)
  - 3D torus (8×6×8)

- **Performance:**
  - 400 TFlops peak
  - 1.3 GB/s EXTOLL bandwidth

- **Two main components**
  - **BNC:** Booster Node Card
  - **BIC:** Booster Interface Card

Thomas Lippert
28.3.2017
Main EXTOLL characteristics
- Direct network: no switches required
- Integrates network interface controller
- Supports 6+1 links
- Capable of tunneling PCIe (allows remote-booting KNC from the network)

Current version of EXTOLL ASIC
- 270 million transistors
- Link bandwidth: 100 G
- MPI latency: 850 ns
- MPI bandwidth: 8.5 GB/s
- Message rate: 70 million mgs/sec
- PCIe Gen3 x16
HW development
Booster Interface Card

Functions

• Boot and manage 16 Intel Xeon Phi
  – Enables Booster Nodes to work autonomously
  – Uses a server (Juno) to address memory addresses of the BNCs over the EXTOLL network

• Interface between Cluster and Booster
  – Has both InfiniBand and EXTOLL
  – Bridges between networks

• Additionally
  – Monitoring functionality for BNCs
  – Lights out management
DEEP Experimental System

DEEP Booster

DEEP Cluster
• **Cluster:**
  - Fat-tree network (InfiniBand)
  - High single thread perf.
  - Large memory per core

• **Booster:**
  - 3D torus network (EXTOLL)
  - Low single thread perf.
  - Few memory per core

The DEEP Software stack should

• allow for an easy programing of this new architecture
• hide underlying hardware complexity
• provide familiar programming environment
• include tools to analyze/optimize application perf.
Application Startup

- **Application's main()**-part runs on Cluster-nodes (CN)
- Actual spawn done via Global MPI
- OmpSs acts as an abstraction layer
- **Spawn** is a collective operation of Cluster-processes
- Highly scalable code-parts (HSCP) utilize multiple Booster-nodes (BN)
Application running on DEEP

Source code

Compiler

Application binaries

DEEP Runtime

```
int main(int argc, char *argv[]){
  /*...*/
  for(int i=0; i<3; i++){
    #pragma omp task in(...) out (...) onto (com, size*rank+1)
    foo_mpi(i, ...);}}
```
DEEP Co-Design Applications

- Brain simulation (EPFL)
- Space weather simulation (KULeuven)
- Climate simulation (CYI)
- Computational fluid engineering (CERFACS)
- High temperature superconductivity (CINECA)
- Seismic imaging (CGGVS)
- Human exposure to electromagnetic fields (INRIA)
- Geoscience (BADW-LRZ)
- Radio astronomy (Astron)
- Oil exploration (BSC)
- Lattice QCD (UREG)
Climate Simulation

Base Model

Grid Points

Data

Exchange

Atmospheric Chemistry

VOC

NO

NO₂

OVOC

Deposition

28.3.2017

Thomas Lippert

45
Modular Supercomputing
Generalization of the Cluster-Booster concept

Module 1: Storage
- Disk

Module 2: Cluster
- CN
- BN

Module 3: Many core Booster
- BN

Module 4: Memory Booster
- NAM
- MEM
- NIC

Module 5: Data Analytics
- DN

Module 6: Graphics Booster
- GN
- NIC

28.3.2017
Thomas Lippert
Advancing Modular Supercomputing with DEEP and DEEP-ER Architectures

By Sean Thielen

February 24, 2017
A BOYFRIEND FOR JURECA
JURECA Cluster
A 2 PFs Multi-Core General Purpose Cluster

- T-Platforms V210 blade server
- Dual-socket Haswell
- ParaStation
- Mellanox InfiniBand EDR
- A few NVIDIA K40 and K80
- Peak: 1.8 PF + 430 TF
- 280 TiB main memory
- 100 GBps storage bandwidth
JURECA Booster
A 5 PFs Many-Core Highly Scalable Cluster

- Intel + DELL
- KNL
- ParaStation
- OmniPath
- Mellanox-Omnipath-Network Bridge → Intel
- ParaStation Cluster-Booster process management (PSPM)
  - PSD → Deamon
  - PSS → Scheduler
  - PSMPI → MPI
- 5 Petaflop/s
Perspectives

2018/19/20

German leadership program “GAUSS” starts second round

- **Jülich Plans:**
  - Bridge Gap between general purpose computing and high scalability
  - Enter Pre-Exascale Era
  - Adapt to deep learning
  - Include Quantum Optimizer

Successor of JUQUEEN will be modular system

2018: GP Cluster Module
2019: HS Booster Module
2020: Data Analytics Module