

## Seminar Final Presentation

# Algorithm Selection and Auto-Tuning in AutoPas

**Manuel Lerchner**  
[manuel.lerchner@tum.de](mailto:manuel.lerchner@tum.de)

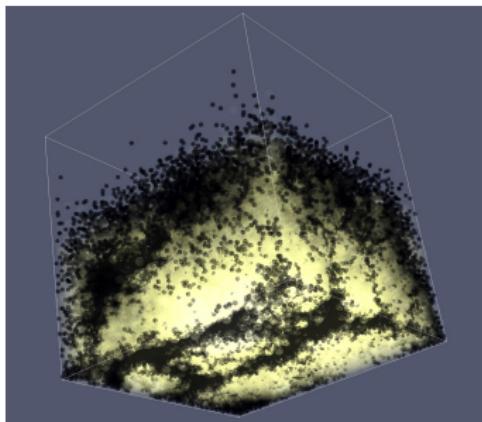
## Table of Contents

## Software Challenges in Molecular Dynamics

- Enormous numbers of particles
  - naively:  $\mathcal{O}(N^2)$  interactions
- Complex interaction models
- Solution: highly optimized algorithms
  - Make full use of hardware
  - Smart data structures

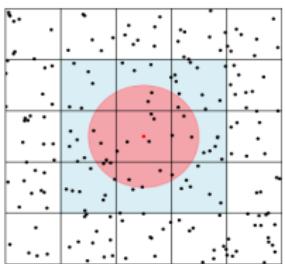
$$E_{ijk} = v \frac{1 + 3 \cos(\theta_i) \cos(\theta_j) \cos(\theta_k)}{(r_{ij} r_{jk} r_{ik})^3}$$

Axilrod-Teller Potential

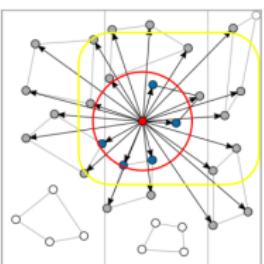


## Background: Particle Containers Optimization

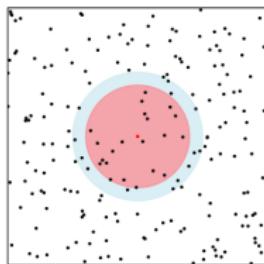
- How to identify interacting particles?
- Short-range forces allow for cutoff radius  $\rightarrow \mathcal{O}(N)$  interactions
- Different implementations possible
  - Unfortunately: No silver bullet



Linked Cells



Verlet Cluster Lists



Verlet Lists

Simpler Memory Access

Lower Memory Overhead

Fewer redundant calculations

## Background: Traversal Patterns

- How to calculate forces in parallel?
- Different traversal patterns possible
- Each pattern has different performance characteristics

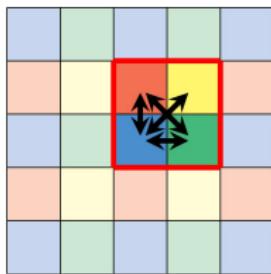
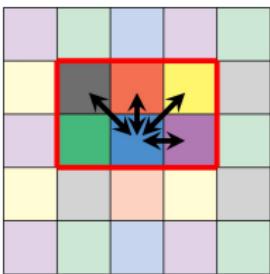
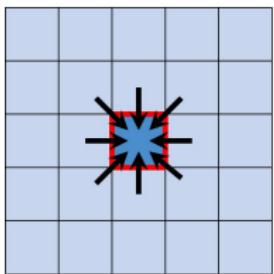


Figure: [?]

## Background: Data Layouts

- How to store particle data in memory?
- SoA
  - Easy SIMD manipulation
- AoS
  - Efficient full particle access

AoS																	
xyz [ ]																	
SoA																	
x [ ]	x <sub>0</sub>	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>	x <sub>5</sub>	x <sub>6</sub>	x <sub>7</sub>									
y [ ]	y <sub>0</sub>	y <sub>1</sub>	y <sub>2</sub>	y <sub>3</sub>	y <sub>4</sub>	y <sub>5</sub>	y <sub>6</sub>	y <sub>7</sub>									
z [ ]	z <sub>0</sub>	z <sub>1</sub>	z <sub>2</sub>	z <sub>3</sub>	z <sub>4</sub>	z <sub>5</sub>	z <sub>6</sub>	z <sub>7</sub>									

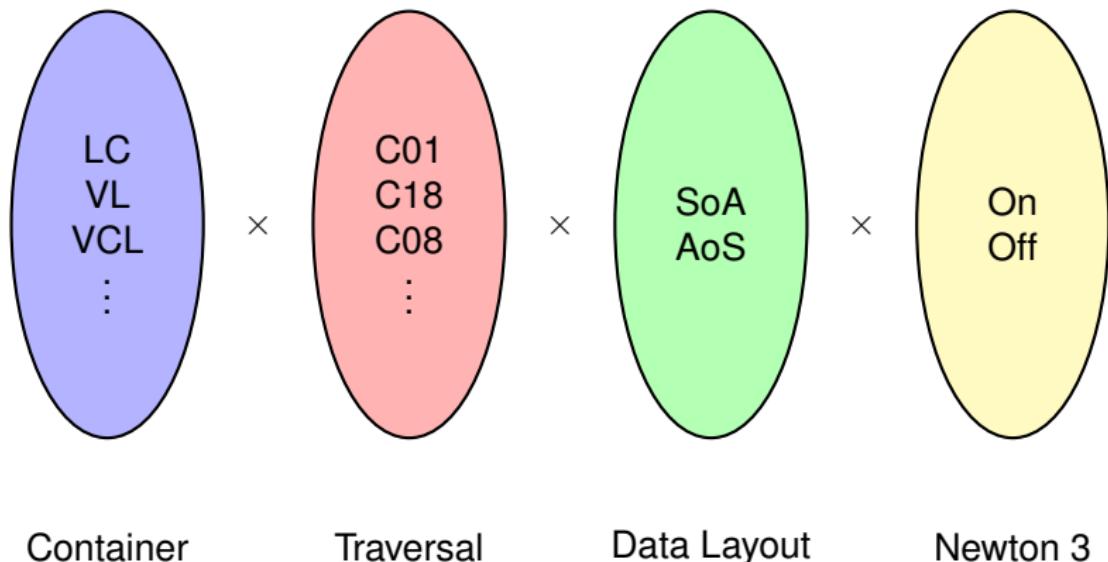
Figure: [?]

## Background: Newton 3 Optimization

- How to optimize force calculations?
- Newton 3: Forces are equal and opposite
- Reuse calculated forces also in other directions
- But: Can lead to race conditions

## Which Combination to Choose?

- Remember: No silver bullet!



Container

Traversal

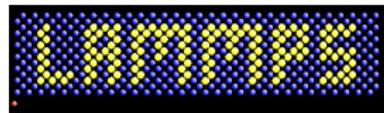
Data Layout

Newton 3

## Traditional MD Engines



- Based on a single container type
  - GROMACS: Verlet Cluster List
  - LAMMPS: Verlet List
  - ls1 mardyn: Linked Cells
- Performance through:
  - Hardware aware SIMD vectorization
  - Memory layout optimization
  - Manual tuning to target hardware
- Drawbacks:
  - Suboptimal for some scenarios / states
  - Manual tuning effort



**ls1**  
Mardyn

## Table of Contents

# AutoPas

## What is AutoPas?

- Library for arbitrary N-body simulations
- Implements all algorithms
- Idea: Switch between algorithms
  - Guarantee optimal performance
- Tuning Strategies:
  - Find optimal configuration
  - Different strategies available

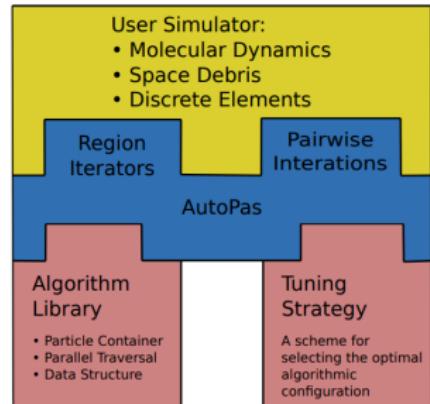
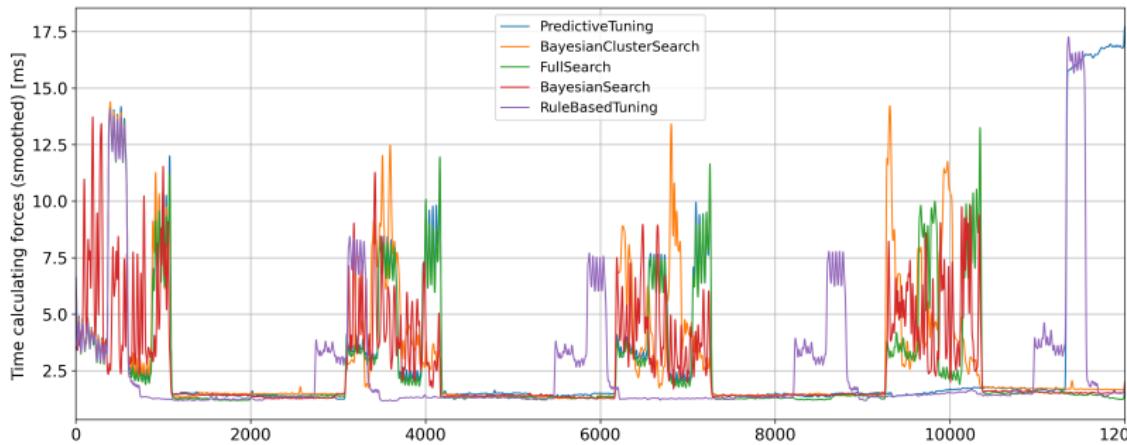


Figure: [?]

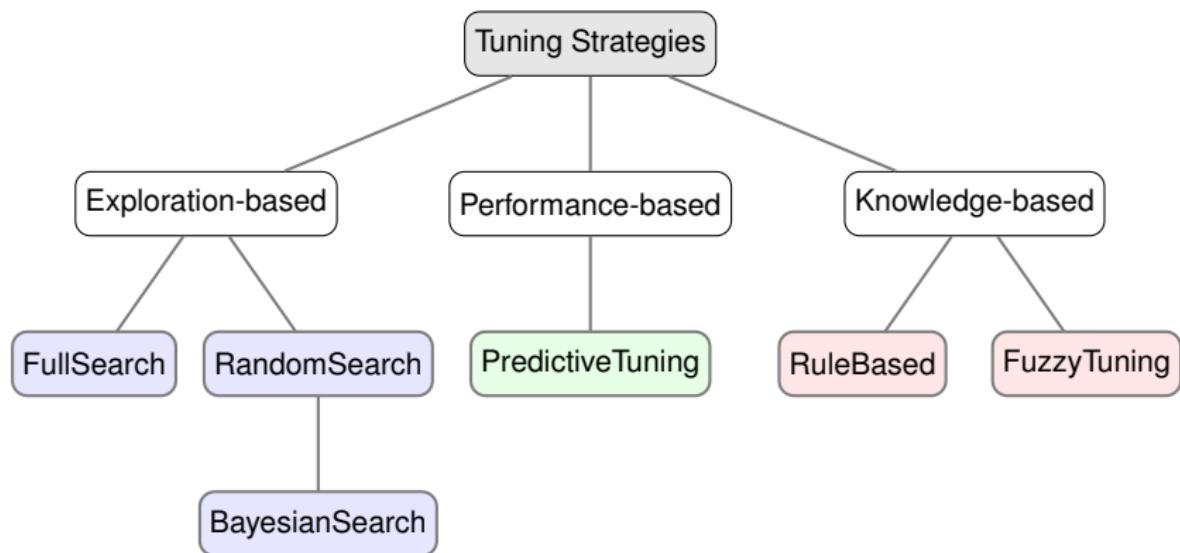
## Table of Contents

## Auto-Tuning in AutoPas

- Divide simulation into Tuning- and Simulation-Phase
- Tuning-Phase:
  - Tuning Strategies suggest configurations
  - Measure promising configurations
  - Choose best configuration



## Tuning Strategies

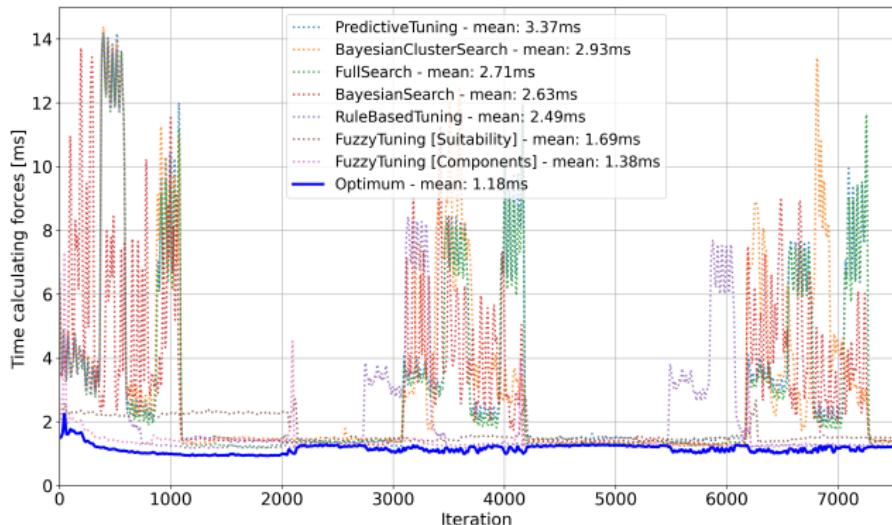


## Benefits of Auto-Tuning

- Optimal configuration
  - for every scenario
  - for every hardware
  - throughout the simulation
- Shown to improve performance in various scenarios
- No manual tuning effort
- AutoPas: Optimal Black-Box for N-body simulations

## Problems of Auto-Tuning

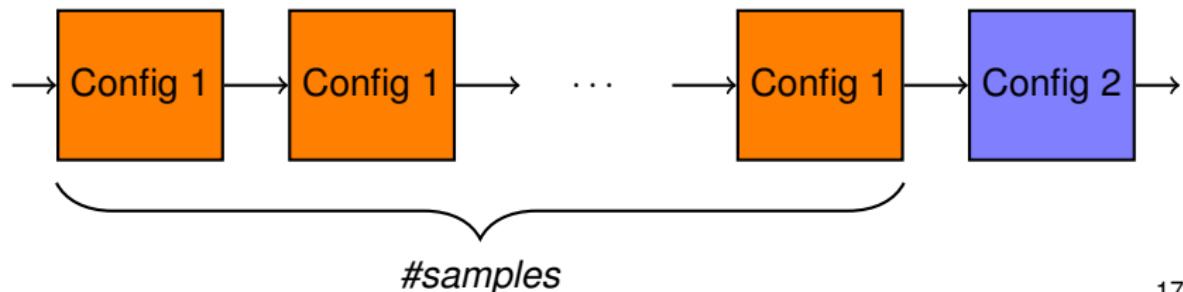
- Overhead from evaluating suboptimal configurations
  - Orders of magnitude slower!
- Unnecessary periodic re-tuning



## Table of Contents

## Early Stopping Idea

- Each configuration gets evaluated multiple times
- Optimization: Abort resampling early if performance is poor
- Two possible strategies:
  - Stop after first poor sample
  - Stop during evaluation

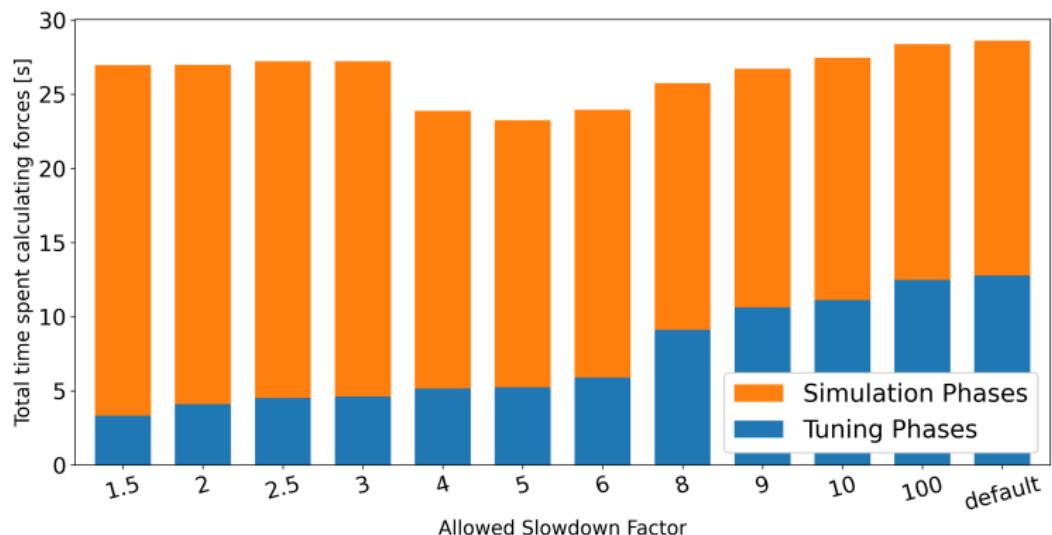


## Naive Early Stopping

- Changes to AutoPas are minimal
  - Keep track of best performance seen so far
  - $slowdownFactor = \frac{t_{\text{sample}}}{t_{\text{best}}}$
  - Call `retune()` if  $slowdownFactor$  exceeds threshold
- Hyperparameter:  $allowedSlowdownFactor$
- Which  $allowedSlowdownFactor$  should be used?
  - $allowedSlowdownFactor \rightarrow 1$ : Many spurious aborts
  - $allowedSlowdownFactor \rightarrow \infty$ : No early stopping

## Benchmark: Exploding Liquid + Predictive Tuning

- Optimal at  $allowedSlowdownFactor = 5$
- From 28.6s to 23.2s
  - 18.9% reduction



## Early Stopping Insights

- Should be combined with good tuning strategy
- Never increased simulation time
- *allowedSlowdownFactor* probably not universal
- Future work needed
  - More benchmarks
  - Improved early stopping mechanism
  - Investigate other stopping criteria

## Summary

- Many algorithmic choices in MD simulations
  - No silver bullet!
- AutoPas: Optimal Black-Box for N-body simulations
  - Auto-Tuning very beneficial
  - But: Overhead from tuning
- Early Stopping Optimization
  - Minimal changes to AutoPas
  - Promising improvements

**Thank you for your attention!**

**Questions?**

## References I

-  Lewis, P. (2020).  
Chapter 03: Modern architectures.
-  Newcome, S. J., Gratl, F. A., Muehlhaeuser, M., Neumann, P., and Bungartz, H.-J. (2024).  
Autopas: Dynamic algorithm selection in molecular dynamics for optimal time and energy.  
In *SIAM Conference on Parallel Processing (PP24)*. SIAM.
-  Newcome, S. J., Gratl, F. A., Neumann, P., and Bungartz, H.-J. (2023a).  
Towards auto-tuning multi-site molecular dynamics simulations with autopas.  
*Journal of Computational and Applied Mathematics*, 433:115278.

## References II

-  Newcome, S. J., Gratl, F. A., Neumann, P., and Bungartz, H.-J. (2023b).  
Towards the smarter tuning of molecular dynamics simulations.  
Amsterdam, The Netherlands.

## **Temporary page!**

$\text{\LaTeX}$  was unable to guess the total number of pages correctly. As there was some unprocessed data that should have been added to the final page this extra page has been added to receive it. If you rerun the document (without altering it) this surplus page will be removed, because  $\text{\LaTeX}$  now knows how many pages to expect for this document.