



SCHOOL OF COMPUTATION, INFORMATION
AND TECHNOLOGY

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

**Exploring Fuzzy Tuning Technique for
Molecular Dynamics Simulations in
AutoPas**

Manuel Lerchner



SCHOOL OF COMPUTATION, INFORMATION AND TECHNOLOGY

DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics

Exploring Fuzzy Tuning Technique for Molecular Dynamics Simulations in AutoPas

Remove all
TODOS

Untersuchung von Fuzzy Tuning Verfahren für Molekulardynamik-Simulationen in AutoPas

Author: Manuel Lerchner

Supervisor: Univ.-Prof. Dr. Hans-Joachim Bungartz

Advisors: Manish Kumar Mishra, M.Sc. &
Samuel Newcome, M.Sc.

Date: 10.08.2024

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, 10.08.2024

Manuel Lerchner

Contents

Acknowledgements	vii
Abstract	ix
Zusammenfassung	xi
1 Introduction	1
1.1 A	1
2 Theoretical Background	2
2.1 Molecular Dynamics	2
2.1.1 Quantum Mechanical Background	3
2.1.2 Classical Molecular Dynamics	4
2.1.3 Numerical Integration	4
2.1.4 Potential Energy Function	4
2.1.5 Simulation Loop	5
2.2 AutoPas	6
2.2.1 Autotuning in AutoPas	6
2.2.2 Tunable Parameters	7
2.2.3 Tuning Strategies	11
2.3 Fuzzy Logic	12
2.3.1 Fuzzy Sets	13
2.3.2 Fuzzy Logic Operations	14
2.3.3 Linguistic Variables	16
2.3.4 Fuzzy Logic Rules	16
2.3.5 Defuzzification	18
4 Proof of Concept	26
4.1 Creating the Knowledge Base	26
4.2 Decision Trees	26
4.3 Fuzzy Decision Trees	27
4.4 Converting a Decision Tree into a Fuzzy Inference System	27
4.5 Creating Fuzzy Rules for <code>md_flexible</code>	32
4.5.1 Data Collection	32
4.5.2 Data Analysis	33
4.5.3 Speedup Analysis	33
4.5.4 Creating the Fuzzy Rules	33
4.5.5 Individual Tuning Approach	34
4.5.6 Suitability Approach	38

5 Comparison and Evaluation	40
5.0.1 Exploding Liquid Benchmark (Included in Training Data)	40
5.0.2 Spinodal Decomposition Benchmark MPI (Related to Training Data)	41
6 Future Work	43
6.1 Better Data Collection	43
6.2 Verification of Expert Knowledge	43
6.3 Future Work on Tuning Strategies	43
7 Conclusion	44
7.1 A	44
A Appendix	45
A.1 Glossary	45
A.2 LiveInfoLogger Data Fields	46
A.3 TuningData Fields	47
A.4 Scenarios used for Data Generation	47
A.5 Data Analysis	47
Bibliography	54

5 Comparison and Evaluation

In this section, we compare the fuzzy tuning technique with other tuning techniques present in AutoPas and evaluate its performance.

To measure the performance of the fuzzy tuning strategy, we also use the scenarios present in `md_flexible` and compare the results with the other tuning strategies present in AutoPas. The benchmarks are run on the CoolMUC-2¹ cluster and are repeated with 1, 12, 24, and 28 threads. We use the `timeSpentCalculatingForces` metric to evaluate the performance of the tuning strategies as it gives a good indication of the overall performance of the simulation.

5.0.1 Exploding Liquid Benchmark (Included in Training Data)

The exploding liquid benchmark simulates a high-density liquid that expands outwards as the simulation progresses. As the data of this scenario was included in the training data, we expect the fuzzy tuning technique to perform well. We only include the benchmark results with one thread for brevity, as the results for the other thread counts are very similar.

The plot in Figure 5.1a shows the time spent calculating the forces for each tuning strategy throughout the simulation. The fuzzy tuning strategies typically perform close to optimal and are very stable. All other tuning strategies show a much higher variance caused by testing many configurations during the tuning phases.

The low tuning overhead is the most significant contributor to the performance of the fuzzy tuning strategies. As the tuning phases of the fuzzy tuning strategies are very short and mainly consist of evaluating already known suitable configurations, there is no overhead caused by the tuning phases. This contrasts with the classical tuning strategies, which spend significant time in the tuning phases.

To show this in more detail, we also include a boxplot of the time spent calculating the forces for each tuning strategy based on the current phase in Figure 5.1b. All tuning strategies show similar timings during the simulation phases, as they eventually found a perfect configuration during the tuning phases but differ drastically in the tuning phases. The fuzzy tuning strategies have a much lower median time spent during tuning phases, with the individual tuning approach performing best. We see that the suitability approach performs worse than the other strategies during simulation phases because the suitability approach chose a suboptimal configuration for the first simulation phase, which was then corrected from the second tuning phase onwards. All other strategies eventually found a perfect configuration during the tuning phases, which caused them to perform better during the simulation phases.

¹CoolMUC-2 is a supercomputer located at the Leibniz Supercomputing Centre in Garching, Germany. It consists of 812 Haswell-based nodes with 14 cores each. As a result of hyperthreading, each node supports up to 28 threads. More information can be found at <https://doku.lrz.de/coolmuc-2-11484376.html>

This plot also shows that the interquartile range of the classical tuning strategies is very similar, with all having nearly identical means. However, all of them are plagued by massive outliers, sometimes taking ten times longer than the median configuration and up to 100 times longer than the optimal configuration. Those extremely bad configurations are the main reason for the poor performance of the classical tuning strategies.

The last plot in Figure 5.1c shows the total time spent calculating the forces for each tuning strategy, again divided into simulation and tuning time. The fuzzy tuning strategies have the lowest total time, with practically no time spent in the tuning phases. Both fuzzy tuning approaches perform similarly and are by far the best-performing strategies. All other strategies typically spend more than 50% of their time in tuning phases where they potentially encounter very bad configurations, which causes them to perform much worse than the fuzzy tuning strategies.

5.0.2 Spinodal Decomposition Benchmark MPI (Related to Training Data)

The spinodal decomposition benchmark simulates an unstable liquid that separates into two phases, each having different characteristics. To improve the performance of the simulation, we used four different MPI ranks, each running on 14 threads to simulate the scenario. As the complete spinodal decomposition benchmark was included in the training data, and the scenario is very homogeneous, we expect the fuzzy tuning strategies to also perform well in this scenario, even if there are no direct training data points for the division of the simulation into multiple MPI ranks.

For brevity, we only include the benchmark results for the 0th MPI rank, as the results for the other MPI ranks are nearly identical.

The plot in Figure 5.2a shows the time spent calculating the forces for each tuning strategy throughout the simulation. This time, we see a difference in both fuzzy tuning strategies, as the component tuning approach performs way better than the suitability approach for most of the simulation. By looking at the boxplots in Figure 5.2b, we see that the suitability approach has the lowest median time spent during the tuning phases. However, it struggles to find the optimal configurations.

Currently, the rule file for the suitability approach specifies that only the top 10% of configurations with the highest suitability should be selected. Increasing this threshold and spending more time in the tuning phase may be beneficial, as currently, the most considerable slowdown is caused by the inability to find the optimal configuration. From Figure 5.2c, we can confirm that the suitability approach spends the least time in the tuning phases, and it could be very worthwhile to increase this limit to improve the total performance.

Figure 5.2 shows that the component tuning approach again performs best, with a decrease in simulation time of around 30% compared to the FullSeach- and BayesianOptimization-based tuning strategies. The suitability and predictive tuning approaches perform similarly and are in second place. Remarkably, the suitability approach performed reasonably well despite never finding the optimal configuration, mainly due to basically no time wasted during the tuning phases. This shows the importance of efficient tuning phases, as they can cause tremendous overhead if not done correctly.

5 Comparison and Evaluation

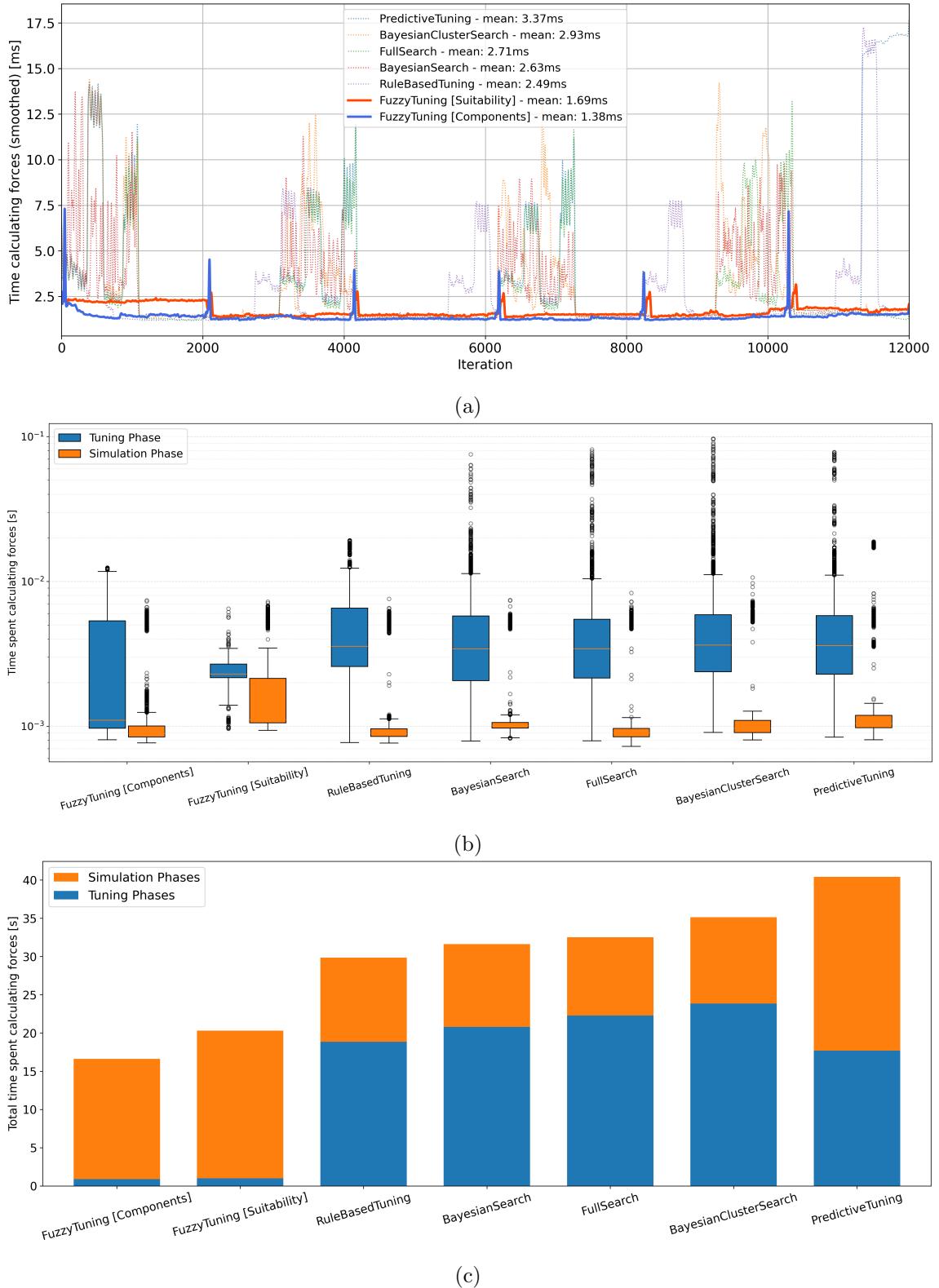
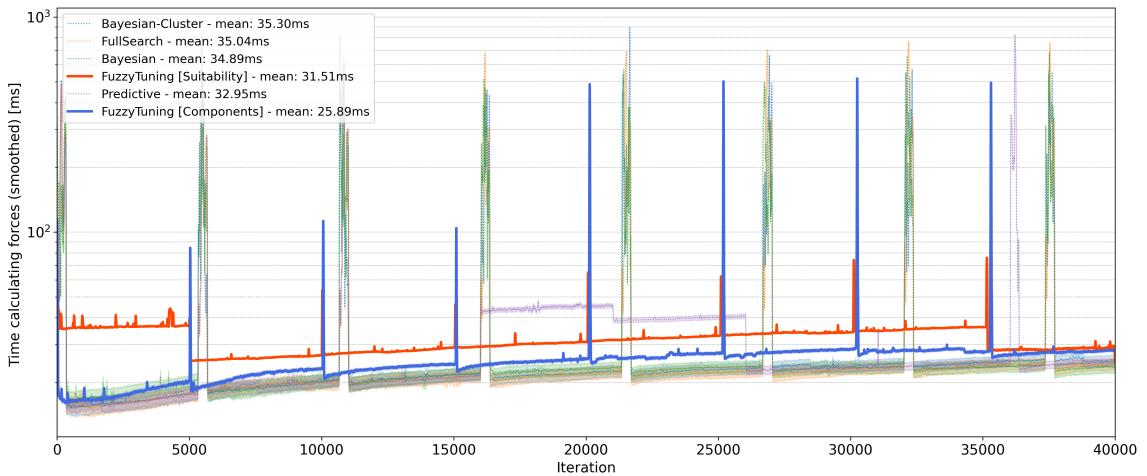
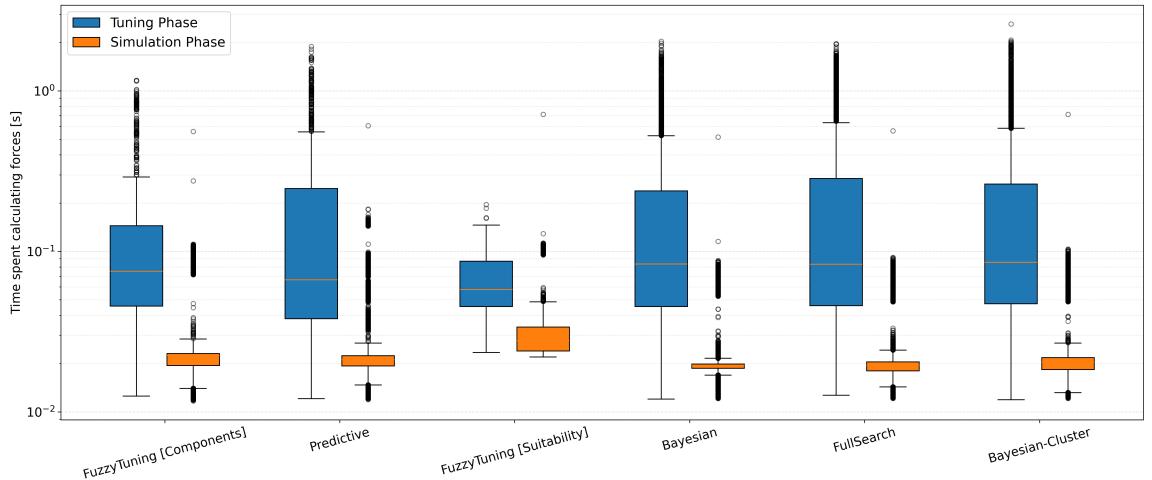


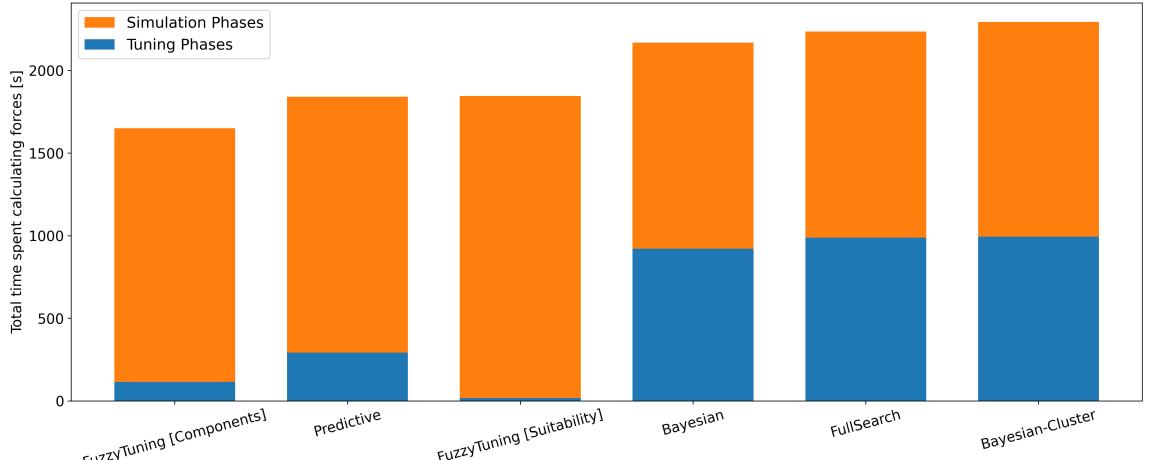
Figure 5.1: Exploring liquid benchmark with 1 thread. (a) Time spent calculating forces. (b) Boxplots of time spent calculating forces divided into simulation and tuning phases. (c) Total time spent calculating forces



(a)



(b)



(c)

Figure 5.2: 0th Rank of the Spinodal decomposition benchmark (Total: 4 MPI ranks, 14 threads each). (a) Time spent calculating forces. (b) Boxplots of time spent calculating forces divided into simulation and tuning phases. (c) Total time spent calculating forces

List of Figures

2.1	MD simulation with 64,423,983 atoms of the HIV-1 capsid. Perilla et al. [PS17] investigated properties of the HIV-1 capsid at an atomic resolution.	2
2.2	MD simulations of shear band formation around a precipitate in metallic glass, as demonstrated by Brink et al. [BPR ⁺ 16].	2
2.3	Visualization of different container options. Source: Gratl et al. [GSBN21]	8
2.4	Visualization of different color-based traversal options. Source: Newcome et al. [NGNB23]	10
2.5	Example of fuzzy sets for the age of a person. Fuzzy sets can be used to model the gradual transition between age groups. The distributions could be derived from survey data on how people perceive age groups. In this example, most people would consider a person middle-aged if they are between 35 and 55; there are, however, outliers ranging as low as 20 and as high as 70.	13
2.6	Effect of different t-norms on the intersection of two fuzzy sets \tilde{A} and \tilde{B} . We can see that the choice of t-norm significantly affects the shape of the resulting fuzzy set.	16
4.1	Decision tree used for the example	27
4.2	Decision surface of the example decision tree	27
4.3	Conversion of crisp tree node into fuzzy tree node	28
4.4	Fuzzy decision tree created from the regular decision tree	29
4.5	Linguistic variables for the converted fuzzy decision tree	29
4.6	Fuzzy inference system created from the fuzzy decision tree seen as a black box	30
4.7	Resulting Fuzzy Set after applying the Rules on specific Data, COG Method	31
4.8	Resulting Fuzzy Set after applying the Rules on specific Data, MOM Method	31
4.9	Decision surface of the fuzzy rules using COG method	31
4.10	Decision surface of the fuzzy rules using MOM method	31
4.11	Speedup density plot of all configurations	34
4.12	Scatterplot of the ContainerDataLayout parameter	37
4.13	Linguistic variable for the homogeneity attribute	37
4.14	Linguistic variable for the Newton3 attribute	37
4.15	Linguistic variable for the Suitability attribute	38
5.1	Exploding liquid benchmark with 1 thread	42
5.2	Spinodal decomposition benchmark MPI with 14 threads	43
A.1	Boxplot of the collected Dataset	48
A.2	Correlation Matrix of the collected Dataset	48
A.3	Speedup density plot based on the Newton 3 option	48
A.4	Speedup density plot based on the Traversal option	49

A.5 Speedup density plot of Configuration-Datalayout option	49
---	----

List of Tables

2.1	Common t-Norms and corresponding t-Conorms concerning the standard negation operator $\neg x = 1 - x$	15
4.1	Extracted fuzzy rules from the fuzzy decision tree	30
4.2	Prepared training data for the Individual Tuning Approach	35
4.3	Extracted fuzzy rules for the Individual Tuning Approach	36
4.4	Prepared training data for the Individual Tuning Approach	39
4.5	Extracted fuzzy rules for the Suitability Approach	39

Listings

Bibliography

- [BMK96] Bernadette Bouchon-Meunier and Vladik Kreinovich. Axiomatic description of implication leads to a classical formula with logical modifiers: (in particular, mamdani's choice of "and" as implication is not so weird after all). 1996.
- [BPR⁺16] Tobias Brink, Martin Peterlechner, Harald Rösner, Karsten Albe, and Gerhard Wilde. Influence of crystalline nanoprecipitates on shear-band propagation in cu-zr-based metallic glasses. *Phys. Rev. Appl.*, 5:054005, May 2016.
- [CBMO06] Keeley Crockett, Zuhair Bandar, David Mclean, and James O'Shea. On constructing a fuzzy inference framework using crisp decision trees. *Fuzzy Sets and Systems*, 157(21):2809–2832, 2006.
- [GSBN21] Fabio Alexander Gratl, Steffen Seckler, Hans-Joachim Bungartz, and Philipp Neumann. N ways to simulate short-range particle systems: Automated algorithm selection with the node-level library autopas. *Computer Physics Communications*, 273:108262, 2021.
- [GST⁺19] Fabio Alexander Gratl, Steffen Seckler, Nikola Tchipev, Hans-Joachim Bungartz, and Philipp Neumann. Autopas: Auto-tuning for particle simulations. In *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 748–757, 2019.
- [LM15] Benedict Leimkuhler and Charles Matthews. *Molecular Dynamics: With Deterministic and Stochastic Numerical Methods*. Interdisciplinary Applied Mathematics. Springer, May 2015.
- [Mur12] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.
- [NGNB23] Samuel James Newcome, Fabio Alexander Gratl, Philipp Neumann, and Hans-Joachim Bungartz. Towards auto-tuning multi-site molecular dynamics simulations with autopas. *Journal of Computational and Applied Mathematics*, 433:115278, 2023.
- [PS17] Juan R. Perilla and Klaus Schulten. Physical properties of the hiv-1 capsid from all-atom molecular dynamics simulations. *Nature Communications*, 8(1):15959, 2017.
- [SGH⁺21] Steffen Seckler, Fabio Gratl, Matthias Heinen, Jadran Vrabec, Hans-Joachim Bungartz, and Philipp Neumann. Autopas in ls1 mardyn: Massively parallel particle simulations with node-level auto-tuning. *Journal of Computational Science*, 50:101296, 2021.

- [VBC08] G. Viccione, V. Bovolin, and E. Pugliese Carratelli. Defining and optimising algorithms for neighbouring particle identification in sph fluid simulations. *International Journal for Numerical Methods in Fluids*, 58(6):625–638, 2008.