

# Managing sustainable and community-driven research software in physics

Jean-Noël Grad, Alexander Reinauer,  
Rudolf Weeber, Christian Holm

Institute for Computational Physics, University of Stuttgart

September 15, 2022



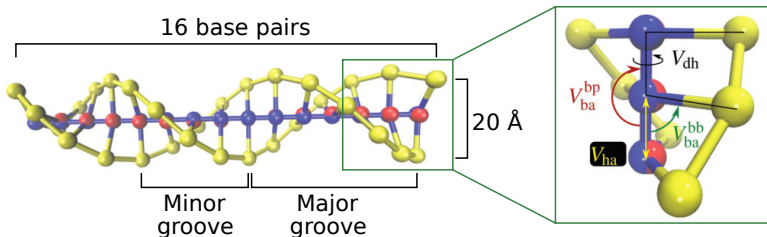
Funded by

**DFG**

Deutsche  
Forschungsgemeinschaft  
German Research Foundation

# Particle-based simulations with coarse-grained models

- ESPResSo represents collections of atoms as single beads
- suitable for systems in which chemical detail can be ignored



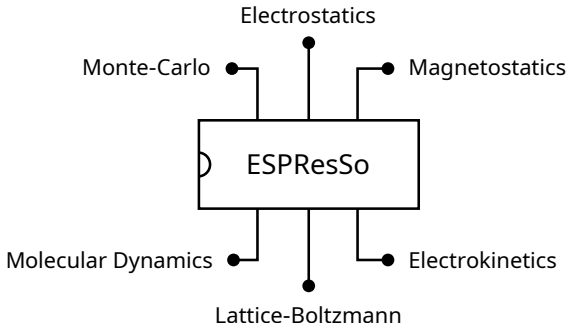
# Computational fluid dynamics with lattice-Boltzmann

- established method to solve Navier-Stokes equations
- ESPResSo introduces coupling to particles
- advanced lattice-Boltzmann features:
  - diffusion-advection-reaction models
  - multiphase flow

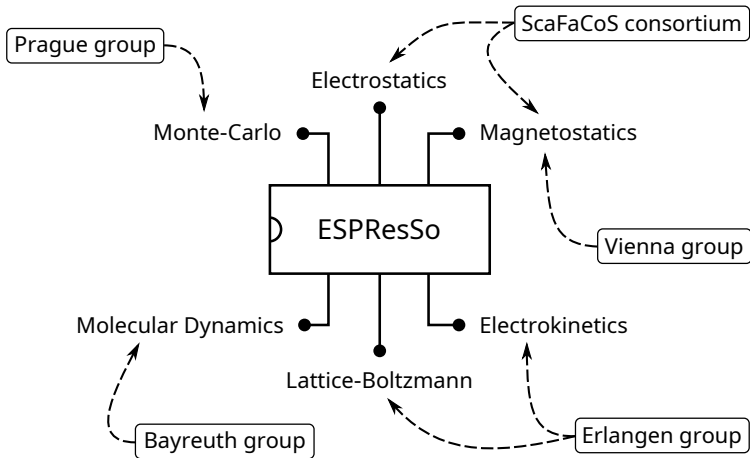


(G. Rempfer, C. Holm; ink droplet in a carrier fluid moving around a solid obstacle)

# Community-driven code development



# Community-driven code development












# Development processes

## Test-driven development

- new feature → write test on day 1
- enforced by continuous integration
  - tests on macOS, Ubuntu, Fedora
  - 99% code coverage is required
- merging is blocked if CI fails

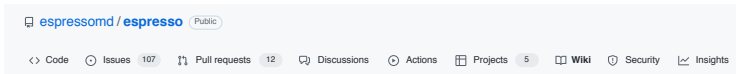
## Documentation-driven development

- Doxygen, Sphinx, Jupyter notebooks
- documentation is tested in CI...
- ...and automatically deployed online

All checks have passed		
7 successful, 1 skipped, and 1 neutral checks		
✓	 run tests on mac / regular_check	<a href="#">Details</a>
⌚	 run tests on mac / sanitizer_check	<a href="#">Details</a>
■	 kodiakhq: status — (no automerge label)	<a href="#">Details</a>
✓	 ICP GitLab CI	<a href="#">Details</a>
✓	 WIP — Ready for review	<a href="#">Details</a>
✓	 codecov/patch — 99% of diff hit	<a href="#">Details</a>
✓	 codecov/project/core — 99% (+0%)	<a href="#">Details</a>
✓	 codecov/project/python_tests — 99% ...	<a href="#">Details</a>
✓	 codecov/project/unit_tests — 99% (+0...	<a href="#">Details</a>

# Onboarding

- wiki: good coding practices, tooling, processes
- online coding days every 3 months
  - pair programming and mentoring
  - collaboration with external contributors



## Development

### Table of Contents

- [Writing code](#)
  - [Source code structure](#)
  - [Build System](#)
  - [Adding new source files](#)
- [Applying formatting locally](#)
- [Testing](#)
- [Debugging](#)
- [Benchmarking](#)
  - [Running individual benchmarks](#)
  - [Running a benchmark suite](#)

### Pages 22

- [Installation FAQ](#)
- [Developer's guide](#)
  - [Tooling](#)
  - [Development](#)
  - [Documentation](#)
  - [Testing](#)
  - [Software design](#)
  - [Naming conventions](#)
  - [Code review](#)
  - [Filing bug reports](#)
  - [Release checklist](#)
- [ESPReso meeting](#)
- [Planned releases](#)
  - [4.3.0 release notes](#)

# Outreach

- annual 5-day CECAM school in October
  - train the next generation of contributors
  - discover ongoing projects in the community
- GitHub discussions, mailing list, community calls
- public notes from developer meetings



MENU

Flagship School

## Simulating the dynamics of soft matter with ESPResSo, PyStencils and LbmPy

October 10, 2022 - October 14, 2022

CECAM-DE-SMSM, Institute for Computational Physics, University of Stuttgart, Stuttgart

Description

Participants

Program

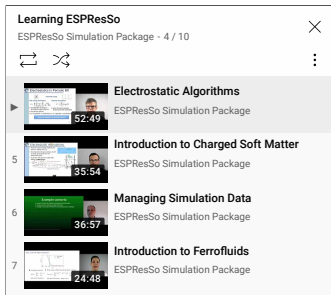
Documents

Participate



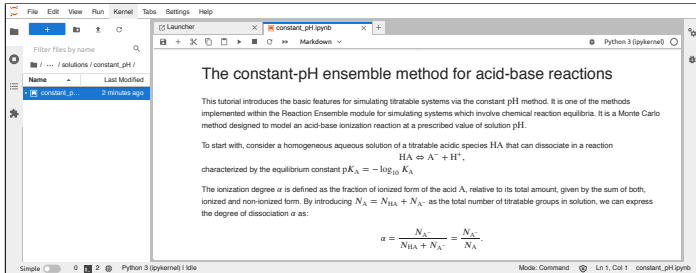
# Dissemination

- packaging on Fedora, Ubuntu
- Singularity for HPC (planned)
- build environment on Gitpod
- run in the browser with Binder
- remotely executable teaching material, lectures on YouTube



**Learning ESPResSo**  
ESPResSo Simulation Package - 4 / 10

- ▶ **Electrostatic Algorithms**  
ESPResSo Simulation Package  
52:49
- 5 **Introduction to Charged Soft Matter**  
ESPResSo Simulation Package  
35:54
- 6 **Managing Simulation Data**  
ESPResSo Simulation Package  
36:57
- 7 **Introduction to Ferrofluids**  
ESPResSo Simulation Package  
24:48



File Edit View Run Kernel Tabs Settings Help

Filter files by name

... / solutions / constant\_pH /

Name	Last Modified
constant_p...	2 minutes ago

## The constant-pH ensemble method for acid-base reactions

This tutorial introduces the basic features for simulating titratable systems via the constant pH method. It is one of the methods implemented within the Reaction Ensemble module for simulating systems which involve chemical reaction equilibria. It is a Monte Carlo method designed to model an acid-base ionization reaction at a prescribed value of solution pH.

To start with, consider a homogeneous aqueous solution of a titratable acidic species HA that can dissociate in a reaction

$$\text{HA} \rightleftharpoons \text{A}^- + \text{H}^+,$$

characterized by the equilibrium constant  $\text{p}K_{\text{A}} = -\log_{10} K_{\text{A}}$

The ionization degree  $\alpha$  is defined as the fraction of ionized form of the acid A, relative to its total amount, given by the sum of both, ionized and non-ionized form. By introducing  $N_{\text{A}} = N_{\text{HA}} + N_{\text{A}^-}$  as the total number of titratable groups in solution, we can express the degree of dissociation  $\alpha$  as:

$$\alpha = \frac{N_{\text{A}^-}}{N_{\text{HA}} + N_{\text{A}^-}} = \frac{N_{\text{A}^-}}{N_{\text{A}}}.$$

Simple 0 2 Python 3 (ipykernel) | Idle Mode: Command Ln 1, Col 1 constant\_pH.ipynb

# Acknowledgments

- ESPResSo core team: A. Reinauer, R. Weeber, F. Weik, K. Szuttor
- Collaborators: S. Bindgen, P. M. Blanco, M. Holzer, D. Mostarac
- Funding: DFG Research Software Sustainability grant 391126171
  
- Gitpod integration: J. Scheunemann
- Jupyter consulting: A. Seeland
- Fedora packaging: C. Junghans



Funded by

**DFG**

Deutsche  
Forschungsgemeinschaft  
German Research Foundation