

Gabriel Pedde Ungureanu | Dottorando in Fisica | HPC & AI

Brescia, Italia

Email: gpeddeun@sisa.it • Web: gabriel-pedde.github.io/Gabriel-Pedde
GitHub: [Gabriel-Pedde](#) • LinkedIn: [gabriel-pedde-ungureanu-4388371ab](#)

Profilo

Dottorando in fisica teorica alla SISSA (6 pubblicazioni peer-reviewed, ~60 citazioni) e matematico (110/110 e lode), iscritto anche al Master in High Performance Computing and AI di SISSA/ICTP. Lavoro dove la matematica incontra il calcolo su larga scala: solver GPU-native in CUDA, calcolo distribuito in MPI e reinforcement learning per LLM su scala EuroHPC.

Esperienza di Ricerca e Sviluppo

SISSA

Trieste, Italia

Dottorando di Ricerca in Fisica Teorica delle Particelle

2022–2026

- Risultati esatti e non perturbativi in teoria quantistica dei campi tramite **calcolo simbolico e numerico su larga scala** (dualità tra teorie di gauge supersimmetriche, indici superconformi) sul cluster EuroHPC Ulysses; 6 pubblicazioni, ~60 citazioni, presentate a conferenze internazionali.
- Pubblicazione sulla **complessità quantistica olografica** (Phys. Rev. D, 28 citazioni), all'interfaccia tra informazione quantistica e gravità.
- Sviluppo di un sistema **multi-agente basato su LLM** ("AI co-scientist": LangGraph, RAG, MCP) per la scoperta scientifica in Relatività Generale, arbitrato da un **valutatore simbolico/numerico deterministico** — il modello non si autovaluta mai: ricostruisce metriche note (Schwarzschild, Kerr, Taub–NUT) con **precisione macchina**.

Formazione

SISSA / ICTP — Programma congiunto

Trieste, Italia

Master in High Performance Computing and AI (MHPC)

2025–2026

In parallelo all'ultimo anno di dottorato. Programmazione GPU (CUDA, OpenACC), calcolo parallelo (MPI/OpenMP), metodi numerici per PDE, architettura degli elaboratori, machine learning, reinforcement learning e calcolo quantistico.

Università Cattolica del Sacro Cuore

Brescia, Italia

Laurea Triennale in Matematica e Laurea Magistrale in Fisica

2017–2022

Analisi reale e funzionale, probabilità, algebra e teoria dei numeri; teoria quantistica dei campi, meccanica statistica. Tesi magistrale in **teoria della complessità quantistica**, all'interfaccia tra olografia e informazione quantistica, pubblicata su Physical Review D.

Premi e Pubblicazioni

6 articoli peer-reviewed in fisica teorica (~60 citazioni) — [INSPIRE-HEP](#) | [Google Scholar](#)

110/110 e lode — Laurea Magistrale in Fisica (**miglior laureato dell'anno**) | 110/110 e lode — Laurea Triennale in Matematica | Borsa di studio al merito Istituto Toniolo

Progetti Selezionati

Solver CFD su GPU: Solver **Lattice Boltzmann (D2Q9, BGK)** scritto da zero in **CUDA**: ristrutturazione AoS → SoA per accessi coalescenti e tiling in shared memory. **1174 GFLOP/s** e **809 GB/s** di banda sostenuta, ~120× rispetto alla baseline CPU ottimizzata su griglia 2048×2048; validato sul decadimento analitico di Taylor–Green | [GitHub](#)

Fine-tuning RL di LLM: Implementazione **da zero di RLOO/PPO** per il fine-tuning di **Llama-3.2-1B** su GSM8K con reward rule-based e regolarizzazione KL: **+11 punti percentuali** di accuratezza (35% → 46%), conformità di formato dal 51% al 100%. Distribuito su **Leonardo (CINECA)**. Stack: PyTorch, TRL, SLURM | [GitHub](#)

Solver di Schrödinger: Risolutore agli autovalori parallelo per l'equazione di Schrödinger stazionaria in **2D/3D** (PETSc DMDA + **SLEPc**, differenze finite distribuite in MPI): spettro dell'idrogeno con **accuratezza <1%** sugli stati eccitati, gap di banda di Kronig–Penney validato contro la teoria degli elettroni quasi liberi | [GitHub](#)

PDE non lineari e ML: Equazione parabolica non lineare stiff (flusso di rinormalizzazione di Wetterich) risolta per due vie indipendenti — **differenze finite** con metodo delle linee (RK45 adattivo) e una **physics-informed neural network** (PyTorch Lightning) — con punto fisso localizzato per **shooting**: esponente critico $\nu = 0,649$ contro 0,630 da Monte Carlo | [GitHub](#)

Competenze Tecniche

Calcolo (HPC): CUDA, cuBLAS, OpenACC, MPI, OpenMP, SLURM, PETSc/SLEPc; analisi di scaling e roofline; Leonardo (CINECA), Ulysses (SISSA), LUMI, Argo (ICTP)

AI / Machine Learning: PyTorch, Hugging Face Transformers e TRL, scikit-learn; reinforcement learning (RLOO, PPO, GRPO, RLHF), CNN, autoencoder, PINN, processi gaussiani, inferenza bayesiana

LLM e agenti: Fine-tuning, RAG, MCP, LangGraph, LangChain, CrewAI, Ollama, Claude Code

Metodi numerici: Differenze finite, volumi finiti ed elementi finiti, solver di Krylov e agli autovalori, PDE non lineari, Monte Carlo

Programmazione: Python (NumPy, SciPy, pandas), C++, C, Fortran, Mathematica, $\text{\LaTeX}$; Git, Docker, Apptainer, CMake, HDF5, Linux

Lingue: Italiano (madrelingua) | Inglese (C1) | Rumeno (A2)