

An Empirical Analysis of High-Performance Computing Education in Germany

Anna-Lena Roth¹  and Jonas Posner¹ 

Fulda University of Applied Sciences, Fulda, Germany
{anna-lena.roth, jonas.posner}@cs.hs-fulda.de

Abstract The increasing relevance of High-Performance Computing (HPC) necessitates the systematic integration of parallel programming and performance-oriented competencies into computational science curricula. Effective HPC education requires not only theoretical foundations but also hands-on experience with real cluster infrastructures to develop a practical understanding of scalability, efficiency, and architectural distinctions between shared and distributed memory systems. Although curricular initiatives advocate such integration, comprehensive cross-institutional evidence on how HPC education is implemented—and how curricular structures relate to locally available infrastructure—remains limited. To examine this relationship, we conducted a systematic empirical assessment of HPC education across 102 academic institutions in Germany. Based on an analysis of module handbooks and course catalogs, 178 HPC-related courses were identified and evaluated with respect to competency coverage and curricular placement. While 67.6% of institutions offer at least one HPC-related course, these offerings are predominantly elective modules situated at the master’s level, with comparatively limited integration in bachelor’s programs. To contextualize these findings, local academic HPC cluster infrastructures were additionally assessed regarding availability, size, and documented accessibility for teaching purposes. Although 61.8% of the institutions operate HPC clusters, only 23.0% explicitly document their availability for educational use, as infrastructures are primarily reserved for research. Statistical analysis reveals a significant association between restricted teaching access and reduced curricular emphasis on practical competencies, including resource management, cluster usage, debugging in parallel environments, and performance analysis. Overall, the findings indicate a structural imbalance between theoretical instruction and the development of practical HPC competencies within German higher education.

Keywords: HPC · Parallel Programming · Education · Curricula

1 Introduction

High-Performance Computing (HPC) has become increasingly important over the past decades. This development is driven by rapidly growing and increasingly precise data volumes in science and industry, including climate simulations, financial

transaction streams, and large-scale training datasets for artificial intelligence. At the same time, computational models have become more sophisticated and accurate, for example, in deep learning and computational fluid dynamics. The combination of large-scale data and complex models results in steadily increasing demands on computational performance and memory capacity.

Modern HPC clusters provide the infrastructure required to distribute computations across thousands of interconnected compute nodes. The performance of such systems is commonly benchmarked and ranked in the Top500 list of the world’s fastest supercomputers [25]. According to this ranking, several powerful systems have reached the exascale level, exceeding 10^{18} floating-point operations per second (e.g., El Capitan [14], JUPITER [16]). However, effectively utilizing HPC clusters requires specialized expertise in parallel programming, cluster architecture, and performance optimization. Consequently, HPC has become an essential component of computer science and computational science education.

Several international initiatives and working groups have proposed guidelines and competency frameworks for structuring HPC curricula [4,15,20,21]. Despite differences in scope and target audience, the initiatives converge on a common set of core competency areas: (1) parallel programming and parallel programming models, (2) HPC theory and architectural understanding, (3) parallel algorithms and mathematical foundations, (4) middleware and resource management, and above all, (5) theory and practice of performance analysis as a pervasive methodological foundation.

Although international curriculum frameworks for HPC education exist, systematic analyses of their implementation remain limited. Most published work reports experiences from individual institutions, resulting in fragmented rather than consolidated insights (e.g., [11,22]). A comprehensive overview of how HPC education is currently structured across German academic institutions is lacking.

This paper addresses this gap by presenting a structured empirical study of HPC education at 102 German *universities* and *universities of applied sciences (UAS)*. The study investigates (1) whether and how HPC is integrated into computer science-related degree programs, (2) which topics are covered within HPC courses, and (3) which infrastructural and institutional resources are available for teaching.

We first describe the study design and data collection methodology in Section 2. The descriptive analysis of the results is presented in Section 3. In Section 4, we contextualize the findings within the international landscape and discuss related work. We then reflect on the implications for HPC education in Section 5, before concluding the paper in Section 6.

2 Data Collection and Methodology

To assess the prevalence and relevance of HPC education at German academic institutions and the content covered in HPC-related courses, we analyzed the module handbooks and course catalogs of 102 institutions offering computer science degree programs. The study was conducted between November and

December 2025 and involved a structured examination of the most recent module handbooks valid for the winter semester 2025/2026. Both bachelor’s and master’s degree programs were included. The sample of 102 institutions was compiled through web-based research to identify accredited computer science programs, and was subsequently verified manually at both the institutional and program level. The final dataset comprises 64 universities and 38 UAS. This distribution reflects the structure of the German higher education landscape, in which universities and UAS represent distinct institutional types with different educational profiles. The raw data underlying this study, including information on the examined institutions, module handbooks, course catalogs, and HPC cluster infrastructures, are publicly available [23].

For the purpose of this study, HPC-related courses are defined as courses that explicitly address programming models, architectures, and tools designed for developing scalable applications on HPC clusters. This includes foundational concepts of HPC cluster architectures, distributed-memory and hybrid systems, communication models, as well as GPU, many-core, and other accelerator technologies. The definition further encompasses courses that provide practical experience with established HPC programming paradigms, including *Message Passing Interface (MPI)* for distributed-memory systems, *OpenMP* for shared-memory systems, and *CUDA* or *Open Computing Language (OpenCL)* for GPU programming. Courses covering parallel numerical methods, scalable algorithms, and performance analysis and optimization on HPC systems were also included. Courses primarily focusing on operating-system-level concurrency or general distributed systems were excluded. In particular, traditional courses on computer architecture (e.g., von Neumann architecture, pipelining, branch prediction), basic thread-based parallelization mechanisms (e.g., POSIX/Java threads, synchronization primitives, semaphores), or operating system internals were not considered unless they explicitly addressed HPC-specific scalability, communication, or performance aspects. Courses that merely utilize HPC clusters (e.g., in machine learning, big data, or cloud computing contexts) without addressing HPC-related programming models or theoretical foundations were also excluded.

For each identified HPC-related course, the following information was systematically recorded:

- Institution name,
- Course title and type (lecture, seminar, lab, etc.),
- Degree program and level (bachelor’s, master’s),
- Elective or mandatory course, and
- Course description and learning objectives.

In total, 178 distinct HPC-related courses were identified through the examination of module handbooks and course catalogs. For these courses, content descriptions and stated learning objectives were extracted and subsequently categorized into recurring HPC topics.

Following topics, derived from the analysis of the course descriptions, served as analytical categories for the subsequent content analysis:

- **MPI**: Use of the Message Passing Interface for distributed-memory parallel programming.
- **OpenMP**: Use of OpenMP for shared-memory parallel programming.
- **GPU**: GPU computing in HPC, including CUDA, OpenCL, or comparable frameworks.
- **Resource Management**: Cluster-based job submission, queue management, and scheduler interaction (e.g., SLURM).
- **HPC Theory**: Fundamental HPC concepts and architectures (HPC clusters, supercomputers, networks, memory hierarchies).
- **HPC Cluster Usage**: Hands-on use of HPC clusters or supercomputers in course activities.
- **Storage Systems**: Parallel storage architectures and file systems (e.g., Lustre [5], BeeGFS [13]), including parallel I/O.
- **Linux/Unix Command Line**: Shell-based interaction with HPC systems, including secure remote access (e.g., SSH), scripting, environment configuration, and batch-oriented workflows.
- **Performance Algorithms**: Parallel algorithms and data structures with performance focus.
- **Performance Theory**: Theoretical runtime analysis, speedup, scalability, and performance models (e.g., Amdahl’s law).
- **Debugging**: Debugging techniques and tools for parallel applications.
- **Performance Practice**: Practical profiling, tracing, and performance optimization (e.g., with Vampir [17]).

In a second step, we analyzed the existing HPC cluster infrastructures at the respective academic institutions. For this purpose, we systematically reviewed the institutional websites to determine whether the institutions operate their own HPC clusters. The following information was collected:

- Name of available cluster(s),
- Operator (e.g., institution computing center, department),
- Hardware information (processors, number of nodes and cores, network),
- Application areas (e.g., research, teaching, special research groups),
- Access conditions (e.g., for staff, students, external users), and
- Affiliation of institutions to HPC consortia.

3 Descriptive Statistics and Results

Table 1 summarizes the results of the analysis of HPC-related courses at German universities and UAS. It reports the number of courses per institution and indicates which of the previously defined HPC topics are explicitly referenced in the corresponding course descriptions. The table provides a high-level overview and does not claim completeness, as actual course content and learning outcomes may differ from the official documentation. Only explicitly mentioned topics were considered. The table also indicates whether each institution operates its own HPC cluster.

University	No. courses	MPI	OpenMP	GPUs	Res. man.	HPC theo.	Clus. usage	Storage	Linux cmd	Algorithms	Perf. theo.	Debugging	Perf. prac.	local clus.
RWTH Aachen Uni.	5	✓		✓		✓		✓		✓	✓	✓	✓	✓
HS Aalen	1			✓		✓					✓		✓	
Uni. Augsburg	2			✓		✓				✓			✓	✓
HS Augsburg	0													
OFU Uni. Bamberg	0													
Uni. Bayreuth	4	✓	✓	✓	✓	✓				✓	✓			✓
Freie Uni. Berlin	1					✓								✓
Humboldt-Uni. Berlin	0													✓
TU Berlin	2	✓	✓				✓							✓
HTW Berlin	0													✓
Uni. Bielefeld	2					✓				✓				✓
RUB Uni. Bochum	2	✓	✓			✓				✓	✓			✓
HS Bochum	1										✓	✓		
RFWU Uni. Bonn	4	✓	✓	✓		✓	✓			✓	✓		✓	✓
TU Brunswick	4	✓	✓			✓			✓	✓	✓			✓
Uni. Bremen	0													✓
HS Bremen	0													
HS Bremerhaven	2	✓								✓				
TU Chemnitz	4	✓		✓		✓	✓	✓		✓	✓			✓
TU Clausthal	1	✓	✓			✓				✓				✓
Brand. TU Cottbus-Senftenberg	0													✓
TU Darmstadt	6			✓	✓	✓		✓		✓	✓	✓	✓	✓
HS Darmstadt	2	✓	✓	✓		✓				✓	✓	✓	✓	✓
TU Dortmund	2			✓		✓				✓				✓
FH Dortmund	1				✓	✓								✓
HHU Uni. Düsseldorf	0													✓
TU Dresden	7	✓	✓	✓	✓	✓	✓			✓	✓		✓	✓
HTW Dresden	1	✓	✓			✓		✓		✓		✓		✓
FAU Uni. Erlangen-Nuremberg	9	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓
FH Erfurt	0													
Uni. Duisburg-Essen	0													✓
HS Esslingen	2	✓	✓	✓						✓	✓	✓	✓	✓
Goethe-Uni. Frankfurt	6	✓	✓	✓	✓	✓	✓	✓		✓	✓			✓
Frankfurt UAS	0													
TU Bergakademie Freiberg	1	✓	✓			✓				✓				✓
ALU Uni. Freiburg	4	✓	✓	✓		✓				✓				✓
HS Fulda	1	✓	✓	✓		✓								✓
JLU Uni. Gießen	1	✓	✓			✓		✓			✓	✓	✓	✓
TH Mittelhessen	3	✓	✓			✓				✓				✓
GAU Uni. Göttingen	4	✓	✓	✓		✓	✓	✓		✓	✓		✓	✓
FernUni. Hagen	3	✓	✓	✓	✓		✓			✓	✓	✓		✓
Uni. Hamburg	3	✓		✓		✓		✓		✓	✓	✓		✓
HAW Hamburg	0													✓
Leibniz Uni. Hanover	2					✓				✓				✓
HS Hanover	1					✓								✓
RKU Uni. Heidelberg	1	✓				✓				✓	✓			✓
SUH Uni. Hildesheim	0													✓
HS Hof	0													
TU Ilmenau	0													✓
TH Ingolstadt	2		✓	✓	✓	✓		✓		✓	✓	✓		✓
FSU Uni. Jena	5	✓	✓			✓				✓	✓		✓	✓
RPTU Kaiserslautern-Landau	1		✓	✓		✓				✓	✓			✓
HS Kaiserslautern	0													
KIT Karlsruhe	9	✓	✓	✓	✓	✓	✓	✓		✓	✓			✓
HS Karlsruhe	0													
Uni. Kassel	3	✓	✓	✓		✓				✓	✓			✓
HS Kempten	1	✓	✓	✓		✓				✓	✓			✓
CAU Uni. Kiel	2	✓	✓	✓		✓	✓		✓	✓				✓

University	No. courses	MPI	OpenMP	GPUs	Res. man.	HPC theo.	Clus. usage	Storage	Linux cmd	Algorithms	Perf. theo.	Debugging	Perf. prac.	local clus.
FH Kiel	0													
Uni. Koblenz	1	✓		✓		✓				✓	✓			
Uni. Cologne	3	✓	✓	✓		✓				✓	✓			✓
TH Cologne	2	✓	✓	✓		✓				✓	✓			✓
Uni. Konstanz	0													
HS Landshut	0													
Uni. Leipzig	1					✓				✓				✓
HTWK Leipzig	0													
Uni. Lübeck	2	✓	✓	✓		✓				✓	✓			✓
TH Lübeck	0													
OVGU Uni. Magdeburg	4	✓	✓	✓		✓	✓	✓		✓	✓	✓		✓
JGU Uni. Mainz	2	✓	✓	✓		✓				✓	✓			✓
Uni. Mannheim	0													
TH Mannheim	3			✓						✓	✓	✓		
PUM Uni. Marburg	0													✓
LMU Uni. Munich	2	✓	✓	✓		✓		✓		✓				✓
TU Munich	8	✓	✓	✓		✓	✓	✓		✓	✓	✓	✓	✓
HS Munich	0													
Uni. Münster	3			✓		✓				✓	✓			✓
FH Münster	1	✓		✓		✓								
TH Nuremberg	0													✓
UOL Uni. Oldenburg	0													✓
Uni. Osnabrück	0													✓
HS Osnabrück	2			✓		✓				✓				✓
Uni. Paderborn	1			✓		✓	✓				✓			✓
Uni. Passau	1	✓	✓								✓			
HS Pforzheim	1	✓	✓	✓				✓			✓			
Uni. Potsdam	4	✓	✓	✓	✓	✓				✓	✓			✓
TH Rosenheim	0													
Uni. Rostock	1	✓	✓			✓					✓	✓		✓
Uni. d. Saarlandes	0													
Uni. Siegen	1	✓	✓			✓								✓
Uni. Stuttgart	4	✓	✓	✓						✓	✓			✓
HFT Stuttgart	1	✓	✓	✓						✓	✓	✓	✓	
Uni. Trier	0													✓
HS Trier	1			✓		✓								
EKU Uni. Tübingen	2			✓		✓				✓	✓		✓	✓
Uni. Ulm	1	✓	✓	✓		✓					✓			✓
Bauhaus-Uni. Weimar	1	✓	✓	✓						✓				
HS RheinMain	0													
HS Wismar	0													
Bergische Uni. Wuppertal	5	✓	✓	✓		✓				✓	✓			✓
JMU Uni. Würzburg	1			✓							✓			✓
HAW Würzburg-Schweinfurt	1	✓	✓	✓		✓					✓			

Table 1: HPC course offerings by academic institution. Thematic areas are aggregated across all courses offered by each institution. ¹

¹ **Abbreviations:** Uni. = Universität; TU = Technische Universität; HS = Hochschule; FH = Fachhochschule; TH = Technische Hochschule; HFT = Hochschule für Technik; HAW = Hochschule für Angewandte Wissenschaften. HTW = Hochschule für Technik und Wirtschaft; HTWK = Hochschule für Technik, Wirtschaft und Kultur; UAS = University of Applied Sciences.

International equivalents: Uni. = University; TU = Technical University; TH = University of Applied Sciences (technical profile); HS/FH/HAW/HTW/HFT/HTWK/UAS = University of Applied Sciences.

Overall, 67.6% of the examined academic institutions offering computer science degree programs provide at least one HPC-related course. Specifically, HPC-related courses are offered by 21 of the 38 UAS (55.3%) and by 48 of the 64 universities (75.0%). Among the 178 courses analyzed, 102 (57.3%) are offered at the master’s level, 39 (21.9%) at the bachelor’s level, and 37 (20.8%) are accessible to students from both levels. The vast majority of courses are electives. Only 31 courses are mandatory for specific degree programs or study specializations. Among institutions that include HPC content in their curricula, the module handbooks contain an average of 2.6 HPC-related courses. UAS that offer HPC provide an average of 1.5 courses, while universities offer an average of 3.1. Notably, FAU Erlangen-Nuremberg (9), KIT Karlsruhe (9), TU Munich (8), TU Dresden (7), TU Darmstadt (6), and Goethe Uni. Frankfurt (6) offer an exceptionally high number of courses and are all universities.

3.1 Content Analysis of HPC Courses

Figure 1 illustrates the percentage distribution of HPC topics identified through the content analysis of the examined course descriptions. HPC theory, including the fundamentals of HPC system architectures and functionality, is covered in 59.6% of the courses. Topics related to the development of parallel algorithms (e.g., sorting algorithms, numerical methods, and matrix multiplications) appear in 56.2% of the courses. Performance-related theory, such as speedup, scalability,

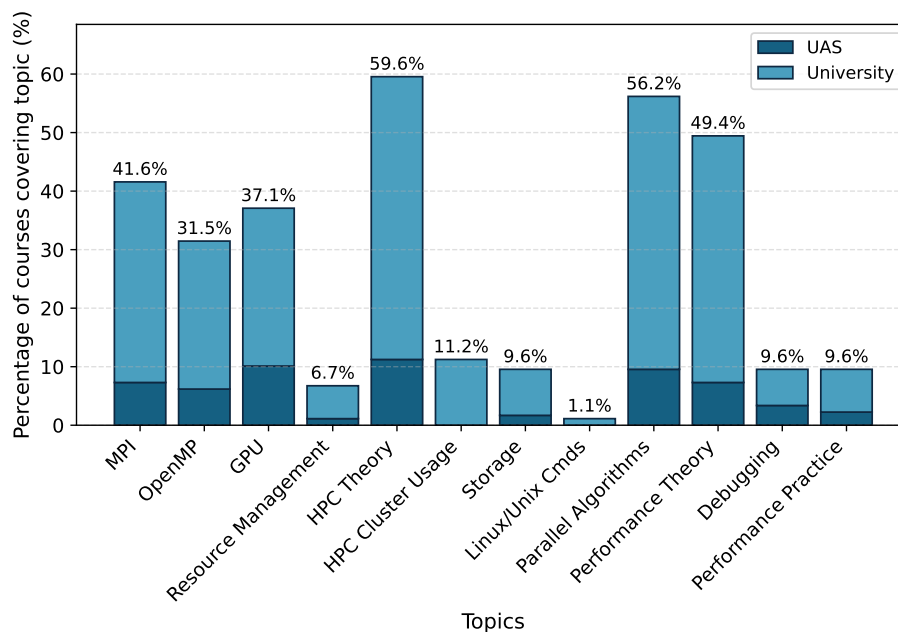


Figure 1: Percentage of HPC courses examined covering HPC topics (n = 178).

and Amdahl’s law, is included in 49.4% of all courses. Parallel programming models are referenced in 41.6% of course descriptions for MPI and in 31.5% for OpenMP, while GPU programming (e.g., CUDA or OpenCL) appears in 37.1%. Some descriptions imply the use of parallel models through references to threads or processes without naming specific frameworks. Thus, the actual prevalence of MPI, OpenMP, or GPU programming may be higher. Explicit HPC cluster usage is documented in only 11.2% of cases. Topics addressed in fewer than 10% of the descriptions include storage systems and parallel I/O (9.6%), parallel debugging (9.6%), practical performance analysis and optimization (9.6%), resource management (6.7%), and Linux/Unix command-line usage in HPC contexts (1.1%). No institution covers all defined HPC topics (see Table 1). FAU Erlangen-Nuremberg and TU Munich cover the largest number of topics (ten out of twelve each) and also offer the highest number of HPC-related courses.

3.2 Analysis of HPC Resources

Of the 102 institutions analyzed, 63 (61.8%) operate at least one local HPC cluster for research and/or education, based on publicly available information [23]. This corresponds to 55 of 64 universities (85.9%) and eight of 38 UAS (21.1%). In total, 87 HPC clusters were identified (79 at universities and eight at UAS). Only clusters that could be explicitly assigned to a specific institution were considered.

Figure 2 shows the distribution of the identified HPC clusters according to their number of CPU cores. Clusters are classified by CPU core count. Exclusively GPU-centric clusters are summarized separately (15 systems).

Eight clusters are classified as small, comprising fewer than 1000 CPU cores (four at UAS and four at universities). The majority of academic HPC clusters fall within the range of 1000 to 4999 CPU cores (17 systems, all at universities). Beyond this range, cluster sizes increase progressively: 13 universities operate systems with 5000 to 9999 cores, ten with 10 000 to 19 999, twelve with 20 000 to 49 999, and ten exceeding 50 000 CPU cores.

Overall, 14 of the 87 identified academic HPC clusters in Germany are listed in the Top500 ranking of the world’s fastest supercomputers (as of November 2025) [25], including five GPU-centric systems. Moreover, nine of the ten clusters with more than 50 000 CPU cores appear in the Top500, indicating a strong correlation between system size and international ranking. These clusters are Goethe-NHR (Goethe University Frankfurt, rank 181), Otus (Uni. Paderborn, rank 191, CPU only), HoreKa (KIT Karlsruhe, rank 202), Emmy (GAU Göttingen, rank 250), Noctua 2 (Uni. Paderborn, rank 309), Fritz (FAU Erlangen-Nuremberg, rank 350), Lichtenberg (TU Darmstadt, rank 390/409, Phase 1/2), CLAIX-2023 (RWTH Aachen, rank 393, CPU only), and Barnard (TU Dresden, rank 419).

Most clusters are primarily intended for research purposes. Only 20 of the 87 identified clusters (23.0%) have explicit educational use documented. The access conditions for the identified HPC clusters vary significantly: Ten clusters are exclusively available to specific research groups. For 49 clusters, an explicit application tied to affiliation with a research project must be submitted. This could particularly affect students who want to access the clusters for thesis work.

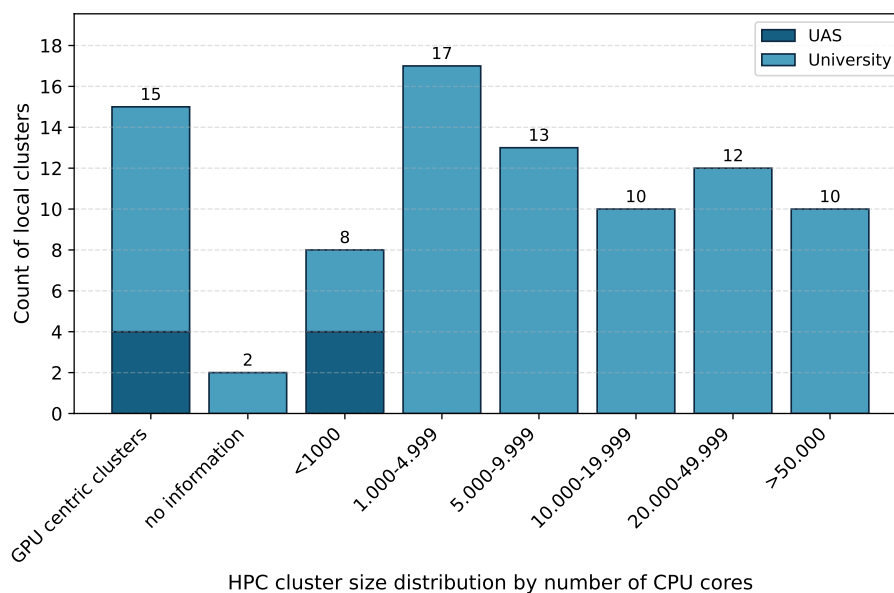


Figure 2: Local HPC clusters at German academic institutions by number of CPU cores ($n = 87$).

For twelve clusters, outside of internal university exceptions, a proposal that undergoes a peer-review process is necessary to gain access. Seven clusters are fully or partially accessible to all members of the respective institutions. For nine clusters, a simple request is sufficient to gain access.

Only a limited number of institutions explicitly document the availability of HPC resources for educational purposes. At BTU Cottbus-Senftenberg, a Beowulf PC cluster is designated for educational use, while the main HPC clusters remain reserved for research. The University of Bonn provides a small GPU cluster and additionally references a Raspberry Pi cluster for instructional purposes. Several universities, including TU Darmstadt, RWTH Aachen, University of Mainz, and TU Dresden, allow limited educational access to partitions of their larger research clusters. The University of Paderborn and HS Fulda make decommissioned systems with older hardware available for teaching.

A notable exception is the *bwUniCluster 3.0* operated by KIT Karlsruhe, which is explicitly designed as a shared educational HPC resource for institutions in Baden-Württemberg. Through the *bwHPC* network [8], this infrastructure is accessible to multiple universities and UAS, including HS Esslingen, RKU Heidelberg, and University of Ulm. Despite these initiatives, 14 of the institutions that operate local HPC clusters do not list any corresponding HPC courses in their official module descriptions.

4 International Comparison

To contextualize the findings on HPC education at German academic institutions, this section places the results in an international perspective by analyzing data derived from the Top500 list. In addition, the findings are compared with international related work on HPC education.

The Top500 list of the world’s fastest supercomputers provides not only performance and technical specifications but also information on the institutional segments to which these systems belong. Within this classification, the academic segment includes systems directly assigned to universities as well as systems operated by national research centers that provide access to multiple academic institutions. For example, the SuperMUC-NG [18], operated at the Leibniz Supercomputing Centre in Munich, is accessible to researchers from various universities through a proposal-based allocation process. Furthermore, the Top500 list may represent previously unified systems as separate entries, for instance, when CPU and GPU partitions of a cluster (e.g., CLAIX-2023 at RWTH Aachen) are listed independently.

Overall, 124 of 500 supercomputers (24.8%) on the Top500 list are assigned to academic institutions worldwide. Thus, approximately $\frac{1}{4}$ of the world’s fastest supercomputers are located at academic institutions or are primarily made available to them. Figure 3 shows the distribution of the academic HPC segments

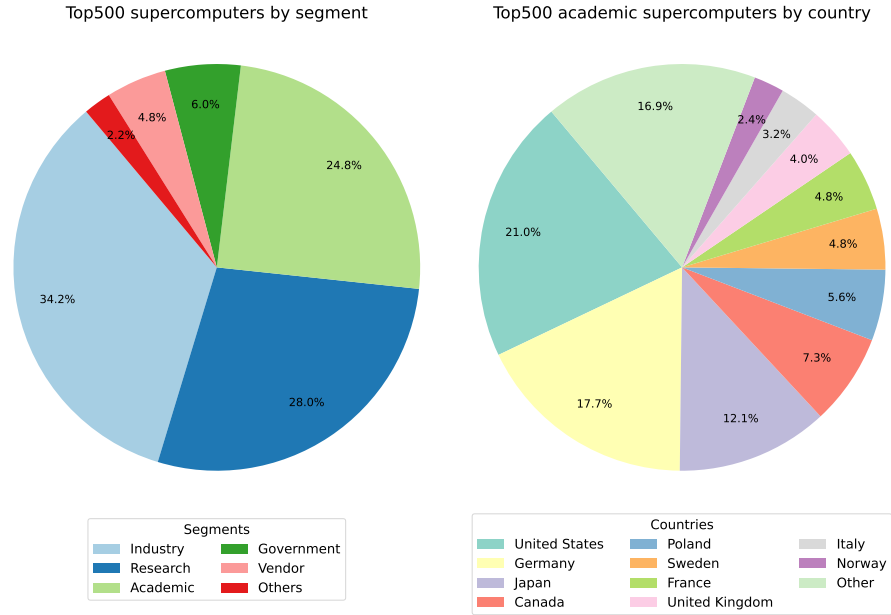


Figure 3: Comparison of HPC segments in the Top500 list (left) and academic supercomputers by country (right). Data source: Top500 [25].

among selected countries. The largest share of the world's fastest academic supercomputers is held by the USA (21.0%), Germany (17.7%), and Japan (12.1%). In total, 25 different countries are represented in the academic category.

In the international context, particularly in the United States, much of the existing literature does not address university-level HPC education but instead focuses on workshops, training programs, and so-called *summer schools* in laboratories [9,10,27]. In the U.S., numerous laboratories and research centers operate HPC infrastructures primarily for research purposes and offer associated training initiatives. These initiatives typically aim to familiarize researchers with specific systems and usage procedures rather than constituting regular, curriculum-embedded university courses. Given their distinct objectives, structures, and target groups, such programs are not considered in the present analysis.

Hebert et al. [12] analyze the availability and use of HPC infrastructure across academic institutions in Wisconsin. They report a strong concentration of HPC resources at large research universities, while smaller institutions rarely operate independent facilities. HPC training is seldom embedded in undergraduate curricula, and active HPC research is largely confined to institutions with local computational resources. Comparing their data with 133 international university-based HPC centers, the authors further show that many centers restrict access to internal users and depend heavily on governmental funding. They also observe a positive correlation between cluster size and dedicated support staff, underscoring the substantial personnel requirements for operating HPC infrastructures.

Samuel et al. [24] examine the use of the Caliburn HPC cluster at Rutgers University by novice users and derive implications for interdisciplinary HPC education. They identify limited resource availability, restricted access, and usability barriers as key structural constraints. In their case study, a first-time user spent the majority of the effort on mastering operational prerequisites rather than on the substantive research task, highlighting the considerable overhead associated with HPC usage. The authors advocate interdisciplinary instructional models and simplified access mechanisms to reduce these barriers.

Al-Jody et al. [1] survey HPC course offerings at eleven UK institutions. Only about half of the analyzed programs include hands-on interaction with HPC clusters, reflecting limited infrastructural capacity. To address this challenge, the authors propose a middleware approach based on container technologies to support instructional computing clusters.

Banchelli and Mantovani [2] investigate HPC curricula at several European universities. While parallel programming models and theoretical performance concepts are consistently covered, practical competencies such as profiling, debugging, energy efficiency, and cluster administration are largely underrepresented. They suggest aligning curricula with competencies required for the Student Cluster Competition, including the integration of performance analysis tools such as *Extrae* [6] and *Paraver* [7].

5 Discussion

Building on the analyses of HPC course content (Section 3.1) and HPC infrastructure availability (Section 3.2), this section synthesizes both dimensions to examine their structural interdependencies and educational implications.

The results indicate that HPC is broadly embedded in German higher education, yet with pronounced institutional asymmetries. HPC appears significantly more frequently in university curricula (75.0%) than at UAS (55.3%), and universities offer, on average, a substantially higher number of HPC courses (3.1 vs. 1.5). This disparity corresponds to infrastructural differences: 85.9% of universities operate local HPC clusters, compared to only 21.1% of UAS. The presence of such infrastructure not only enables practical training but also reflects the availability of specialized personnel capable of supporting HPC education.

HPC-related courses in Germany primarily emphasize theoretical foundations, parallel algorithms, performance theory, and programming with MPI, OpenMP, and CUDA. Offerings are predominantly elective modules at the master’s level, while bachelor-level integration remains limited. The five thematic areas proposed by HPC education working groups and initiatives, which are considered essential for comprehensive HPC education (see Section 1), are primarily reflected in the German curricula we examined. Parallel programming and parallel programming models (1) are well represented, particularly through MPI (41.6%) and OpenMP (31.5%), as well as GPU programming with, e.g., CUDA (37.1%). HPC theory and architectural understanding (2) are covered in 59.6% of all courses. Parallel algorithms and mathematical foundations (3) are addressed in 56.2% of all courses. Middleware and resource management (4) are included only in 6.7% for resource management and 9.6% for storage topics such as I/O and parallel file systems. In the theory and practice of performance analysis and optimization, 49.4% of all courses cover performance theory, whereas only 9.6% include practical performance optimization.

Despite the apparent formal alignment with recommended competency areas, a substantial imbalance between theoretical and practical components emerges. While performance-theoretical topics such as Amdahl’s law and the concepts of speedup, scalability, and load balancing are widely covered, practical skills required to identify and resolve performance problems (e.g., profiling, tracing, optimization) are significantly underrepresented (9.6%). Such competencies are methodologically demanding and typically require access to specialized toolchains and HPC infrastructures, and thus remain difficult to integrate systematically in the absence of appropriate infrastructure. Explicit HPC cluster usage is documented in only 11.2% of courses, and practical resource management (6.7%)—including scheduler-based job submission and allocation via SLURM—as well as debugging in parallel environments (9.6%) are similarly rare. These figures indicate that authentic cluster-based workflows are not consistently embedded in curricula. Linux/Unix command-line proficiency, including shell-based interaction with HPC systems, job scripting, and batch processing, is mentioned in only 1.1% of the analyzed course descriptions. Given that many HPC courses are positioned at the master’s level, such competencies may be assumed as prior

knowledge rather than explicitly taught. However, the overall low representation of cluster-related topics suggests that numerous courses rely on local programming environments instead of sustained hands-on interaction with HPC systems. These findings align with international observations by Banchelli and Mantovani [2], who similarly reported that debugging, complex code optimization, and energy efficiency are underrepresented topics in HPC education.

The infrastructural analysis further reveals a structural accessibility gap. Although 61.8% of the analyzed academic institutions operate HPC clusters—some even represented in the Top500 ranking—only 11.2% of courses explicitly document cluster usage, and merely 23.0% of clusters are documented as accessible for teaching. Access is frequently restricted by proposal-based allocation procedures or limited to specific research groups, creating substantial barriers for educational integration. These patterns are consistent with the findings of Samuel et al. [24], who identify administrative constraints, access restrictions, and usability hurdles as major impediments to effective HPC adoption in academic contexts.

Taken together, the results suggest that HPC infrastructures in Germany are primarily organized as research resources rather than as systematically integrated educational environments. The mere presence of high-performance systems does not automatically translate into accessible, practice-oriented training opportunities. This structural configuration particularly disadvantages UAS, where limited local infrastructure constrains the integration of authentic HPC workloads into teaching.

Shared HPC infrastructures, such as the *bwUniCluster 3.0* in Baden-Württemberg, as well as regional and national consortia including HKHLR [3], HPC.NRW [26], and the nationwide NHR infrastructure [19], may mitigate access disparities between institutions. However, shared systems also introduce governance, scheduling, and administrative constraints that may restrict pedagogical flexibility and institutional autonomy. Coordinated resource allocation, limited configuration privileges, and competing research workloads can reduce the feasibility of sustained hands-on instructional use.

Overall, the findings reveal a systemic tension between research-oriented HPC infrastructures and the requirements of competency-oriented HPC education, resulting in structurally limited opportunities for comprehensive practical skill development.

6 Conclusion

This study presents a systematic analysis of HPC education at German academic institutions, integrating curricular and infrastructural perspectives. While HPC is widely embedded and key thematic areas are formally represented, a clear imbalance persists between theoretical coverage and practice-oriented competencies.

Although many institutions operate HPC infrastructures, limited educational access and the rare integration of authentic cluster-based workflows restrict opportunities for hands-on experience. Practical competencies—particularly Linux

command-line skills, resource management, debugging, and performance analysis—remain underrepresented despite their relevance for research and industry.

The findings indicate a structural misalignment between research-oriented HPC infrastructures and competency-oriented educational objectives. Strengthening practical HPC education, therefore, requires not only curricular refinement but also sustainable, low-barrier access to HPC systems for teaching.

References

1. Al-Jody, T., Aagela, H., Holmes, V.: Inspiring the Next Generation of HPC Engineers with Reconfigurable, Multi-Tenant Resources for Teaching and Research. *Sustainability* **13** (2021). <https://doi.org/10.3390/su132111782>
2. Banchelli, F., Mantovani, F.: Filling the Gap Between Education and Industry: Evidence-Based Methods for Introducing Undergraduate Students to HPC. In: *Proceedings of EduHPC (2018)*. <https://doi.org/10.1109/EduHPC.2018.00008>
3. Bischof, C.: Hessisches Kompetenzzentrum für Hochleistungsrechnen (HKHLR) (2026), <https://www.hkhlr.de/de>
4. Bouvry, P., Brorsson, M., Canal, R., Eftekhari, A., Höfinger, S., Smets, D., Köstler, H., Kozubek, T., Krishnasamy, E., Llosa, J., Lukas-Rother, A., Martorell, X., Pleiter, D., Proykova, A., Sancho, M., Schenk, O., Silvano, C.: The European Master for HPC Curriculum. *Journal of Parallel and Distributed Computing* **201** (2025). <https://doi.org/10.1016/j.jpdc.2025.105081>
5. Braam, P.: The Lustre Storage Architecture (2019), <http://arxiv.org/abs/1903.01955>, arXiv:1903.01955
6. BSC Tools: Extrae (2026), <https://tools.bsc.es/extrae>
7. BSC Tools: Paraver (2026), <https://tools.bsc.es/paraver>
8. bwHPC: High Performance Computing, Data Intensive Computing and Large Scale Scientific Data Management in Baden-Württemberg (2025), <https://www.bwhpc.de>
9. Callaghan, S., Filinger, W., Zhukov, I., Lederer, H., Urbanic, J.: Fifteen Years of International HPC Summer School. In: *Practice and Experience in Advanced Research Computing*. ACM (2025). <https://doi.org/10.1145/3708035.3736011>
10. Diehl, P., Li, Y.W., Junghans, C., Holmen, J.K., MacCarthy, E., Parete-Koon, S., He, Y.H., Hartman-Baker, R., Lively, C., Gott, K., Gupta, L., Streu, K., Ghadar, Y., Kinsley, P., Herriman, J., Draeger, E.W., Eijkhout, V., Mehringer, S.: Shaping the Future Workforce: Challenges and Lessons Learned in HPC Education from National Labs and Computing Centers (2025). https://doi.org/10.35542/osf.io/sua4v_v1
11. Döpfer, T., Große-Wöhrmann, B., Lindner, D., Milakovic, D., Oexle, J., Resch, M.M., Scheel, O., Slotosch, S., Widmaier, L.: Expanding HLRS Academic HPC Simulation Training Programs to More Target Groups. *The Journal of Computational Science Education* (2021). <https://doi.org/10.22369/issn.2153-4136/12/3/2>
12. Hebert, J., Hratish, R., Gomes, R., Kunkel, W., Marshall, D., Ghosh, A., Doss, I., Ma, Y., Stedman, D., Stinson, B., Varghese, A., Mohr, M., Rozario, P., Bhattacharyya, S.: High-Performance Computing in Undergraduate Education at Primarily Undergraduate Institutions in Wisconsin: Progress, Challenges, and Opportunities. *Education and Information Technologies* **29** (2024). <https://doi.org/10.1007/s10639-024-12582-6>
13. Herold, F., Breuner, S., Heichler, J.: An Introduction to BeeGFS. Tech. rep., ThinkParQ, Kaiserslautern, Germany (2014), https://www.beegfs.io/docs/whitepapers/Introduction_to_BeeGFS_by_ThinkParQ.pdf

14. Hewlett Packard Enterprise: El Capitan (2025), <https://www.hpe.com/us/en/compute/hpc/lawrence-livermore-national-laboratory.html>
15. Informationstechnische Gesellschaft im VDE and Gesellschaft für Informatik e.V.: Curriculum für Bachelor- und Masterstudiengänge Technische Informatik (March 2018), <https://dl.gi.de/server/api/core/bitstreams/031111d5-9360-43e7-aed4-54bb34950308/content>
16. Jülich Supercomputing Centre: JUPITER|The Arrival of Exascale in Europe (2025), <https://www.fz-juelich.de/en/jsc/jupiter>
17. Knüpfer, A., Brunst, H., Doleschal, J., Jurenz, M., Lieber, M., Mickler, H., Müller, M.S., Nagel, W.E.: The Vampir Performance Analysis Tool-Set. In: Tools for High Performance Computing - Proceedings of the International Workshop on Parallel Tools for HPC. Springer (2008). https://doi.org/10.1007/978-3-540-68564-7_9
18. Leibniz Supercomputing Centre: SuperMUC-NG (2026), <https://doku.lrz.de/supermuc-ng-10745965.html>
19. NHR-Verein e.V.: Nationales Hochleistungsrechnen (2026), <https://www.nhr-verein.de/>
20. Prasad, S.K., Weems, C.C., Sussman, A., Gupta, A., Estrada, T., Vaidyanathan, R., Ghafoor, S.K., Kant, K., Stunkel, C.B.: NSF/IEEE-TCPP Curriculum on Parallel and Distributed Computing for Undergraduates - V. II - Big Data, Energy, and Distributed Computing. In: Proceedings of the Technical Symposium on Computer Science Education. ACM (2023). <https://doi.org/10.1145/3545947.3569594>
21. Raj, R.K., Romanowski, C.J., Impagliazzo, J., Aly, S.G., Becker, B.A., Chen, J., Ghafoor, S., Giacaman, N., Gordon, S.I., Izu, C., Rahimi, S., Robson, M.P., Thota, N.: High Performance Computing Education: Current Challenges and Future Directions. In: Proceedings of the Working Group Reports on ITiCSE. ACM (2020). <https://doi.org/10.1145/3437800.3439203>
22. Raoofy, A., Elis, B., Bode, V., Chung, M.T., Breiter, S., Schlemmon, M., Herr, D.F., Fuerlinger, K., Schulz, M., Weidendorfer, J.: BEAST Lab: A Practical Course on Experimental Evaluation of Diverse Modern HPC Architectures and Accelerators. The Journal of Computational Science Education (2024). <https://doi.org/10.22369/issn.2153-4136/15/1/5>
23. Roth, A.L.: HPC at German Universities - Data analysis references [Data set] (2025). <https://doi.org/10.5281/zenodo.18722580>
24. Samuel, J., Brennan-Tonetta, M., Samuel, Y., Subedi, P., Smith, J.: Strategies for Democratization of Supercomputing: Availability, Accessibility and Usability of High Performance Computing for Education and Practice of Big Data Analytics (2021), <https://arxiv.org/abs/2104.09091>, arXiv:2104.09091
25. Strohmaier, E., Dongarra, J., Simon, H., Meuer, H.: Top500 The List (2025), <https://top500.org>
26. University Paderborn: HPC.NRW (2026), <https://hpc.dh.nrw.de/>
27. Wofford, J.L., Lueninghoener, C.: The Supercomputer Institute: A Systems-Focused Approach to HPC Training and Education. The Journal of Computational Science Education (2020). <https://doi.org/10.22369/issn.2153-4136/11/1/12>