White Paper: “New Mathematics for the Exascale: Applications to Materials Science”

A long program of the Institute for Pure and Applied Mathematics, UCLA, in Spring 2023

Soumendu Bagchi, Los Alamos National Laboratory
Ilgar Baghishov, The University of Texas at Austin
Maciej Buze, Heriot-Watt University
Weiqi Chu, University of California Los Angeles
Sambit Das, University of Michigan
Tim Germann, Los Alamos National Laboratory
Jason Gibson, University of Florida
Peter Hatton, Los Alamos National Laboratory
Richard Hennig, University of Florida
Thomas Hudson, University of Warwick
Abigail Hunter, Los Alamos National Laboratory
Jan Janssen, Los Alamos National Laboratory
Sung Hoon Jung, The University of Texas at Austin
Jiyoung Lee, The University of Texas at Austin
Julien Luzzatto, Massachusetts Institute of Technology
Jörg Neugebauer, Max-Planck-Institut für Eisenforschung
Amedeo Perazzo, Stanford University
Danny Perez, Los Alamos National Laboratory
Andrew Rohskopf, Sandia National Laboratories
Michael Taylor, Los Alamos National Laboratory
Elin Theilen, Fraunhofer MEVIS
Aidan Thompson, Sandia National Laboratories
Mira Todorova, Max-Planck-Institut für Eisenforschung
Daniel Utt, OVITO GmbH
Ping Yang, Los Alamos National Laboratory
Eva Zaat, University of Warwick
Wei Zhu, University of Alabama

Executive Summary 3
1. Introduction 4
Background 4
Historical perspective 5
2. Workflows for the Exascale 6
Scientific opportunities 7
Current state-of-the-art 7
Outstanding challenges 8
Progress at IPAM 9
Vision statement 10
3. Multiphysics Simulations 11
Scientific opportunities 11
Current state-of-the-art 12
Outstanding challenges 12
Progress at IPAM 13
Vision statement 15
4. Co-Design for Exascale Architectures 15
Scientific opportunities 15
Current state-of-the-art 16
Progress at IPAM 17
Outstanding challenges 19
Vision statement 20
5. Materials Informatics at the Exascale 21
Scientific opportunities 21
Outstanding challenges 22
Progress at IPAM 23
Vision Statement 24
6. Making the Exascale Accessible for its Expected Users 24
Current state-of-the-art 26
Outstanding challenges 27
Progress at IPAM 28
Vision Statement 28
Appendix: A Non-Human Perspective 30

Executive Summary

This document summarizes the activities and outcomes of the Long Program “New Mathematics for the Exascale: Applications to Materials Science” which was held at the Institute of Pure and Applied Mathematics (IPAM) from March 13 to June 16, 2023. It briefly explores some of the current open questions and future directions in the field of exascale computing with applications to materials sciences. The program included a series of tutorials followed by four focused workshops:

- “New Mathematics for the Exascale: Applications to Materials Science Tutorials” provided an introduction to major themes of the entire program, in order to build a foundation for the participants who have diverse scientific backgrounds.
- “Workshop I: Increasing the Length, Time, and Accuracy of Materials Modeling Using Exascale Computing” focused on recent development of new mathematical approaches to intensive calculations at massive scale, particularly new ways to improve scalability (both weak and strong) and extend the size, time, and accuracy of simulations.
- “Workshop II: Scale-Bridging Materials Modeling at Extreme Computational Scales” focused on new mathematical approaches to multiscale/multiphysics modeling, with a particular emphasis on theoretical and numerical challenges faced at the exascale.
- “Workshop III: Complex Scientific Workflows at Extreme Computational Scales” was aimed at developing new mathematical and computational approaches that enable the inclusion of massive-scale computing into complex scientific workflows.
- “Workshop IV: Co-design for the Exascale and IPAM Hackathon” brought together code developers, mathematicians, method developers, computer scientists, and engineers from the computer vendors for a week of discussion, hands-on development, and implementation of the new ideas generated during the program.

In addition to these workshops, several working groups were formed which met regularly throughout the program, focusing on different topics such as electronic structure theory, multi-fidelity long-timescales simulations, model reduction, co-design of molecular dynamics, neural networks for partial differential equations, workflows for exascale computing, and co-design of machine-learned interatomic potentials.

This document is organized around the key topics that were identified by the program’s participants. They reflect areas where we believe the convergence of mathematical advances, new computational science tools and practices, and domain sciences expertise are essential. These topics are:

- *Workflows for the Exascale:* We expect that exascale machines will be heavily used to carry out complex workloads composed of coupled tasks where decision-making will be required at runtime. Specialized tools are required to enable the efficient concurrent execution of such workloads without human intervention.
- *Multiphysics Simulations:* Multiscale/physics/fidelity modeling is an accepted paradigm in materials modeling that could greatly benefit from exascale computing. Automating the

exchange of information between models while retaining control of the uncertainties remains an important challenge.

- *Co-Design for Exascale Architectures*: Upcoming exascale computers rely heavily on hardware accelerators for performance. This places unique constraints on the type of simulations that can efficiently execute on these machines.
- *Materials Informatics at the Exascale:* Exascale computers will generate massive amounts of data, which presents unique opportunities to materials informatics approaches, but also serious challenges.
- *Making the Exascale Accessible for its Expected Users:* The heterogeneity, scale, and complexity of exascale computers present serious challenges to non-expert domain users, such as graduate students.

We note that this document does not aim at providing an exhaustive review of the field. It reflects the subjective opinion of the participants, which was formed throughout the program. As such, references were intentionally omitted.

1. Introduction

Background

Computing resources, such as those provided to the community by the U.S. National Science Foundation (NSF) and Department of Energy (DOE), are exponentially growing, both in capacity and in capability. As we enter the exascale era, the most powerful computing systems will deliver in excess of 10^{18} operations per second. Such a tremendous amount of computer power has the potential to revolutionize the role of computational approaches in the scientific enterprise, e.g., by allowing for the purely *in silico* development of new materials. Yet many (most!) computational scientists do not benefit from this technological revolution and continue to compute at modest scales, or employ parallel computers in a capacity-computing framework, i.e., to carry out a large number of small calculations. Indeed, more than 84% of calculations carried out at NERSC at Lawrence Berkeley National Laboratory used only 1 node, and close to 94% less than 10 nodes. While a few users are able to leverage thousands of nodes, the overwhelming majority uses supercomputers much like they would a departmental cluster.

One of the key reasons for this is that broadly-used computational tools are often unable to scale to massive computational resources. These limitations are not simply due to practical programming issues that could easily be solved; instead, they often *fundamentally stem from the way problems are mathematically posed and solved*. Scalability depends on problem formulations and on solution strategies that are intimately aware of architectural constraints, in terms of data locality, use of memory hierarchies, communication costs, vectorizability, etc. This requires a dramatic rethinking of the types of problems that should be pursued, of how they are mathematically formulated, of how they are solved, and of how these solutions are implemented on actual hardware. Only then can extreme-scale computing make a profound scientific impact on the community. In other words, scalability to extreme scales cannot be an afterthought, but has to be considered from the very infancy of the development of a research program.

As eloquently stated by DOE's Applied mathematics research for exascale computing working group: "*Advances in applied mathematics, in areas such as mathematical modeling, numerical analysis, and adaptive algorithms, will be essential in order to produce high-performance exascale applications and will provide key input to application scientists and computer scientists.*" Further, they point out that addressing this challenge cannot be achieved by the different communities in isolation: "*Domain scientists must work with mathematicians to formulate problems, […]. Mathematicians must work with computer scientists and engineers to develop new algorithms and implementations that can efficiently harness architectural features. Computer scientists must collaborate with domain scientists and mathematicians to ensure that programming environments […] provide functionality relevant to their needs.*"

Our program was designed to address the challenges posed by the exascale in the materials simulation community, as computations there have been playing a critical role for decades, creating a considerable pool of domain expertise.

Historical perspective

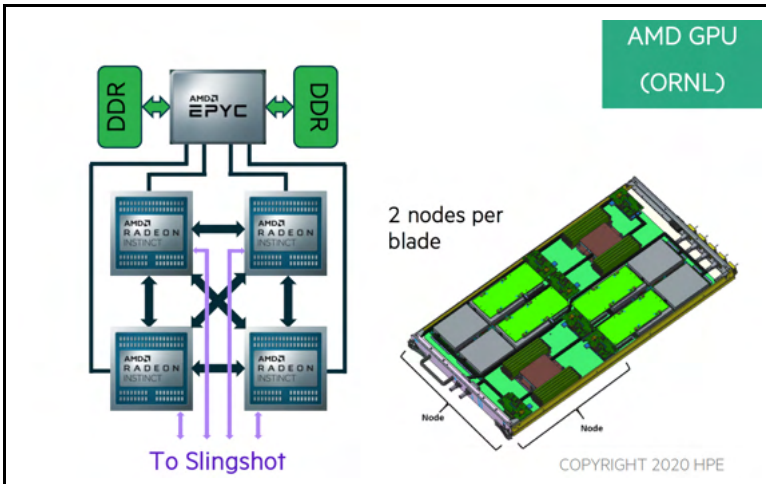

Figure 1: Node diagram on the Frontier exascale supercomputer at ORLN. Each node contains one AMD EPYC CPU and 4 AMD Radeon GPUs, which provide the bulk of the computing power of the system. The almost 10,000 nodes of the machine are connected by a fast interconnect. Note the non-uniform memory access: DDR RAM cannot be directly accessed by the GPUs.

The first petascale supercomputer was Los Alamos National Laboratory (LANL)'s Roadrunner, which also introduced the era of accelerated supercomputing by combining general-purpose CPUs with specialized accelerators. Installed in 2008, Roadrunner was followed by other petascale platforms using the general-purpose graphical processing units (GPGPUs) pioneered by NVIDIA, as well as homogeneous many-core processor-based systems, most notably the IBM BlueGene series. As soon as the petascale era had arrived, a series of workshops and reports focused on the challenges and opportunities for the exascale era that was anticipated to arrive a decade or so later. The common belief at the time was that these two "swim lanes" would compete, and that the first exascale machine would either be based on many-core processors or on GPUs. A 2009 report predicted that a many-core-based exascale machine would require 1,000,000 nodes, while a GPU-based one would require "only" 100,000 nodes. Concurrency, the latency and bandwidth of both memory accesses and internode communication, and resilience were anticipated to be the major challenges.

The exascale era arrived only 14 years later with ORNL's Frontier supercomputer, which achieved 1.102 Exaflop/s in the June 2022 Top500 list. The many-core swimlane had ended a few years earlier, leaving the field open to GPU-based systems. With nodes composed of AMD EPYC CPUs and MI250x GPUs, and a novel Slingshot interconnect. As of this writing (June 2023), Frontier remains the only exascale supercomputer, but is expected to be joined by Argonne National Laboratory (ANL)'s Aurora (with Intel Sapphire Rapids CPUs and Ponte Vecchio GPUs) and LLNL's El Capitan (with AMD MI300A Accelerated Processing Unit (APU), integrating CPUs and GPUs in a single package) by the end of 2024. Each of these systems relied on relatively few ($O(10^4)$) very powerful and heterogeneous nodes, far exceeding the computational performance of two swimlanes predicted more than a decade earlier. The massive multilevel concurrency presents tremendous opportunities for algorithms which can exploit it, but also tremendous challenges to many traditional approaches.

Indeed, exascale architectures pose stringent conditions on scientific applications, which must:

- efficiently leverage GPUs, which deliver the vast majority of the computing power,
- explicitly handle heterogeneity, in the form of hardware accelerators, memory hierarchies, storage hierarchies, etc., and
- carefully balance communication and computation, as computation becomes increasingly cheap compared to data motion.

The goal of this program is to tackle these challenges to learn to leverage the exascale to solve extremely computationally intensive complex, multi-physics, multi-scale problems in materials science, through the development of novel mathematical approaches. This requires progress on the development of scalable single-scale solvers (e.g., molecular dynamics (MD), density functional theory (DFT), discrete dislocation dynamics, phase field, finite element, etc.), of novel ways to couple and bridge the scales (both at the level of mathematical formalisms and of implementation), and of ways to orchestrate massive online workflows and dataflows. Our focus is specifically on *"big-compute",* i.e., on developing new paradigms for computing at scale, and on developing strategies for efficiently harnessing the output of such massive scale simulations.

2. Workflows for the Exascale

Computational workflows in materials science are graph networks of interconnected nodes. These graph nodes can be heterogeneous simulation codes, real-time experimental data analysis, and scientific libraries. Bringing these tools together accomplishes greater outcomes not achievable by any individual software tool alone. With the rise of exascale computing most individual software tools cannot be upscaled to arbitrarily large and heterogeneous computational resources. Rather, divide-and-conquer approaches are required for connecting individual tools to realize multi-scale and multi-physics simulations as well as virtual replicas of real-world materials or systems that mimic their behavior and response to various conditions (digital twins). A computational workflow

allows a direct representation of such applications by connecting and orchestrating the input and output of the individual workflow nodes.

Scientific opportunities

The fundamental challenge of materials science is coupling different time and length scales to predict microscopic and macroscopic material properties which can be used to identify new strategies in materials design. Consequently, workflows in materials science are often intrinsically hierarchical and couple simulation codes developed by different communities. These communities bring in their specific expertise on a specific scale or of a certain physical aspect in highly domain-specific simulation codes. Starting from parameter-free *ab-initio* calculations and propagating results and uncertainties computed by the various simulation codes across scales enables highly accurate quantitative predictions of material properties that often rival experimental measurements. These novel workflows enable inverse materials design and digital twins. To develop workflows that efficiently utilize the resources and capabilities of exascale computers the availability of a formal language that describes the complex simulation protocols is crucial. Such a language which is applicable for both theoretical and experimental workflows should address the following three aspects:

- **Virtual Lab**: It should provide an intuitive and user-friendly interface for developing workflows which naturally and efficiently support the heterogeneous architecture of the up-coming generation of exascale computers.
- **Composability**: It should define standardized and interoperable interfaces for heterogeneous simulation codes and scientific libraries which are easily integrable with the individual components. This would enable reusability of codes, reproducibility of results and avoid repeated implementations of standard operations.
- **FAIR**: It should provide the necessary infrastructure to develop FAIR (Findable, Accessible, Interoperable, and Reusable) workflows and workflow nodes. Such FAIR workflow nodes could be shared directly by multiple workflows, providing code reusability.

While the availability of such a language is not specific to materials science, it is particularly essential for the hierarchical, multi-scale and multi-physics workflows discussed in the Section 3 below, which require the integration and connection of simulation codes that are not inherently interoperable and describe very different domain sciences.

Current state-of-the-art

Currently, material science workflows can be broadly classified into two categories. First, there are high-throughput simulations performed for a single level of physics (e.g. electronic structure) but going over a large number of chemical compositions, structures, material properties etc. Second, there are workflows coupling multiple tools. For both cases, state-of-the-art workflows are typically implemented as either monolithic highly specialized codebases that subdivide the workload across available resources or as collections of scripts that “glue” together the desired tools. The tools are tailored to the specific materials science challenge and the HPC hardware on which they are executed.

From this, workflow systems have recently emerged and they allow computational materials scientists to create useful abstractions, which help to organize and codify such scripts. Examples of generic software frameworks for such abstractions are `Snakemake`, `Dask` and `Parsl`. Within the materials science community, specialized workflow managers and tools have been developed and enabled high-throughput computation, with frameworks such as `ASE`, `pymatgen`, `fireworks`, `Aiida`, and `pyiron`.

State-of-the-art platforms for workflow design tend to be highly linear in execution, typically following *if, then, else* forms in terms of task chains. Examples of this type of task coupling include e.g. machine learning interatomic potential training with iterative schemes flowing from structure generation, to UQ analysis, to high-level theory querying, to re-training of machine learning interatomic potentials.

Outstanding challenges

A major challenge in exascale materials computation is the combination and coupling of simulation codes. Typically, the inputs and outputs of such codes differ semantically but often they are also conceptually different. A recurring issue is coupling one level of theory (e.g. interatomic potentials) with simulations performed at other levels of theory (e.g. electronic structure). Similarly, the increasing complexity of data analysis algorithms brings the need to utilize simulations to refine and drive data processing steps. For example, workflows that combine the analysis of experimental data with simulations to spawn a digital twin of an experiment could be used to distinguish between competing hypotheses in real-time to close the loop between experiment and theory.

Another outstanding challenge is the separation of the workflow graph and the execution of the corresponding tasks and having unified schemes that can handle the asynchronous execution and different levels of parallelization. For example, many materials science workflows combine highly optimized internally-`MPI`-parallelized, serial, and gpu-optimized codes. Scheduling and distributing these workflow tasks among the heterogeneous architecture of exascale machines is critical and remains a challenge for existing workflow frameworks. Workflows in the framework of experimental studies, for example at future light sources, require real-time interpretation of data to inform experimental operations and design, to enable increased knowledge extraction from spectroscopy and scattering studies at light sources. Complex workflows that encompass the experimental and the computational facility, and the network connecting them, will be needed to orchestrate the resources, stream the data, and perform the analysis.

Finally, workflows for exascale machines should efficiently distribute and evaluate their highly diverse tasks across the heterogeneous computational architecture available on a given machine. Thus, support for synchronous task execution independent of the complex simulation protocols has to be provided by next-generation exascale materials workflow frameworks. Furthermore, optimizing the usage of resources requires advanced automated decision-making strategies based on mathematical theory, as discussed in Section 5. Beyond these computational

challenges, experimental workflows additionally face the challenge of being distributed over multiple facilities, which adds an additional level of complexity for coordinating the individual steps of the workflows as well as data motion.

Progress at IPAM

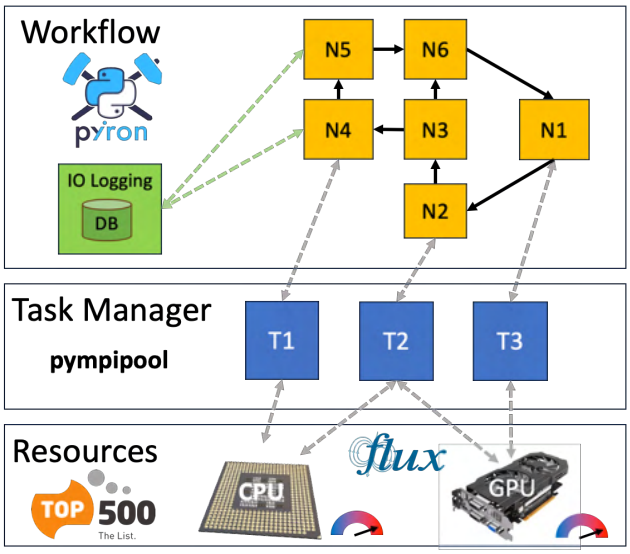

Figure 2: Illustration of a workflow conceptualization as developed at IPAM with (upper) workflows defined as graphs, (middle) a task manager to distribute tasks created by workflow graph nodes (N1 to N6), and (lower) resource manager to distribute tasks (T1 to T3) to the CPU and GPU-based hardware present. The developed approach enables maximum usage of the hardware resources.

To separate the workflow graph and its nodes from the execution of the individual workflow tasks, a task manager has been developed, which distributes tasks over heterogeneous architectures. This task manager handles both serial and parallel simulation codes as well as codes, regardless of whether they provide a programmatic python interface or not. Each task submitted to the task manager is executed asynchronously, providing a full separation of the physics implemented in the workflow graph from the task execution.

This novel interface, which has been developed at IPAM, is based on the `Flux` hierarchical resource manager. In combination with additional developments to support the seamless integration of serial and parallel python tasks these developments resulted in the development of a python library - `pympipool`. This novel interface has been, as part of this IPAM activity, integrated in the `pyiron` framework. To make it easily accessible to up-coming exascale computers, as well as to simplify the migration of existing simulation protocols developed with the `pyiron` framework, a `conda` package has been provided. The interface for the `pympipool` library has been kept general and can be straightforwardly extended to other workflow solutions. It is applicable for both domain scientists as well as mathematicians and provides timings to quantify and predict the computational costs of individual tasks.

Vision statement

Enabling the full potential of exascale computing power requires complex workflows that can combine a multitude of domain-specific simulation tools and mathematical libraries. In the future, it will be increasingly important that the computing codes are optimized for hybrid parallel architectures. Ideally, codes should be interoperable and composable to reduce communication barriers between various tasks in a workflow. This could be realized by computing codes providing the option of being called as libraries rather than requiring reading and writing to file systems. In practice, since code developers come from different communities, it is unlikely that the necessary standardization will happen soon on the level of the code. With workflow frameworks materials science can be significantly advanced by achieving the following short-term and intermediate milestones:

- Design active-learning enabled, resource-aware workflows that queue the tasks in a way that maximizes the knowledge gain at fixed computational costs. Developments in data science and machine learning could leverage this task by utilizing collected statistics of computing cost and knowledge information.
- Generate standardized wrappers around existing scientific codes that extract the key information and bring it into a generic format. The generic input/output format will allow for connecting tools, which originally are not interoperable and without any human intervention to build complex workflows. These wrappers make workflow nodes FAIR, enabling integration into a generic workflow. A crucial aspect of such wrappers is that they need to include comprehensive error handling specific to individual codes.
- Develop uncertainty quantification and computational cost estimation of the individual workflow nodes to enable the speculative execution of workflows to predict knowledge gain.
- Enable publication of FAIR workflows that are reproducible, transferable and extendable.

3. Multiphysics Simulations

Scientific opportunities

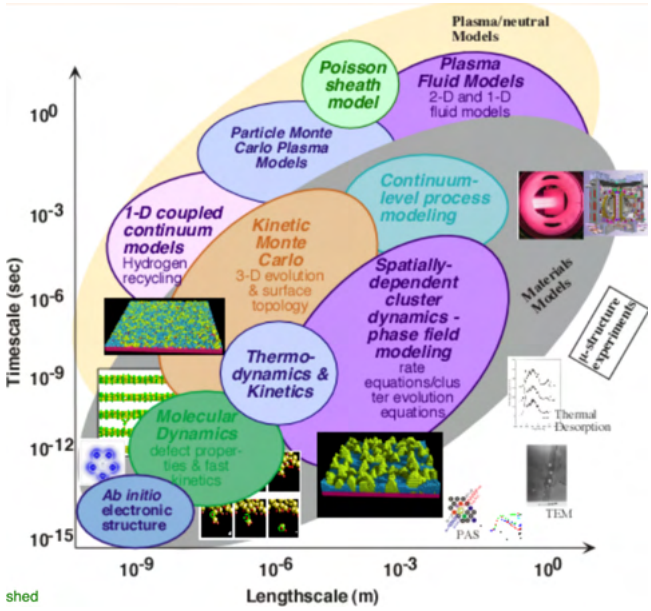

Figure 3: Prototypical illustration of a multiphysics modeling framework for fusion-energy materials. From Wirth and Whyte (unpublished).

The widely embraced Materials Genome Initiative (MGI) framed the critical role of computations in accelerating the design cycle of new materials, which is of considerable importance to economic competitiveness. Indeed, the time between the discovery of a new class of materials in the lab and its deployment in industrial applications averages about 20 years, which makes it difficult to respond to emerging challenges (e.g., clean energy, alternatives to rare earth materials, etc.) , in a timely manner. The US Department of Commerce estimates that reducing this development cycle by half would result in improvements in direct R&D efficiency of about $60B per year, on top of an economic impact of about $150B per year resulting from improved R&D outcomes. The MGI has led to the explosion of high-throughput computing efforts that often consume hundreds of millions of CPU hours per year. While this approach has been very successful for functional materials, where performance can often be inferred from small first-principle calculations, predicting the performance of structural materials (like steels for nuclear applications or lightweight alloys for high-efficiency vehicles) is qualitatively more complex. Indeed, performance prediction is an intrinsically multiscale problem where the structure of the material at the atomic, nano, meso, and micro-scales can be critical. In this case, performance prediction relies on integrating information from simulations carried out at multiple physical scales, with multiple levels of fidelity, capturing different physics; an approach we will generically refer to as multiphysics modeling in the following, although it also encompasses methods that are traditionally referred to as multiscale or multifidelity methods.

Multiphysics modeling is a well-established paradigm in materials modeling and various variants have been demonstrated over the years. E.g., Fig. 3 presents a notional multiphysics approach to the design of plasma-facing materials for future fusion reactors. It combines multiple physics (electronic structure simulations, thermodynamics and kinetics, plasma modeling, etc) operating

at multiple time and length physical scales (from the angstroms to meters and from femtoseconds to minutes). The amount and diversity of calculations that are required to make predictions at engineering scales will clearly require the availability of colossal amounts of computing which makes this type of problem an ideal target for exascale platforms.

Current state-of-the-art

Multiphysics modeling frameworks have been successfully demonstrated for a broad range of applications in the past; see e.g., Fig. 3 for an illustration of an application to fusion materials. Of particular interest to our program are methods such as (i) the use of *ab initio* electronic structure calculation to parameterize interatomic potentials for molecular dynamics, (ii) the use of molecular dynamics to discover reaction pathways for Kinetic Monte Carlo simulations, (iii) the use of Kinetic Monte Carlo simulations to parameterize continuum cluster-dynamics/phase fields models of microstructural evolution of the material. The unifying theme for most of these approaches is that the scale-bridging is primarily carried out offline and sequentially. Manual intervention is typically required to select, set up, execute, and analyze the simulations.

Concurrent multiscale methods, where different methodologies/codes are coupled at run-time (e.g. density-functional theory calculations embedded into classical MD, or classical MD embedded into finite element simulations) are exceptions to this rule. Concurrent approaches are however typically limited to the coupling of a small number of methods. We note that parametric sensitivity analysis of the output of large-scale codes parameterized from lower-scale simulations have also been demonstrated.

Outstanding challenges

While the current-state-of-the-art described above has produced a number of successful outcomes, it suffers from many limitations that precludes the use of massive-scale computing resources to accelerate the design and qualification of new materials:

1. Human intervention is required at every step. As the availability of massive amounts of computing power increases, humans can become the rate-limiting step. It is therefore essential to automate as many steps as possible, ideally completely removing human intervention. The specifics of this idea will be discussed in Section 5, but in general this requires the development of automated methods to manage simulations and to exchange information between the different scales and codes. This automation is key to maximally exposing computational parallelism in the upscaling chain and hence the amount of computing resources that can be efficiently leveraged. To fully realize this, as much of the computational workload as possible should be executed concurrently, enabling the seamless exchange of information. However, enabling efficient concurrency requires addressing the challenge of principled resource allocation between the different tasks so as to minimize the uncertainty on the target quantities of interest. Solving this problem will

require new methods for seamless and scalable uncertainty quantification across scales which are free of poorly justified simplifying assumptions (the i.i.d. Gaussian priors, perfect model specification, conjugate priors, etc.).

2. A key limiting factor in the development of a multiphysics approach is the fact that the representation of the physics in the different formalisms can be difficult to reconcile. Developing mappings between representations typically involves human intervention and the introduction of simplifying assumptions that are often made for computational convenience. These discontinuous changes in representation across scales can introduce errors that are difficult to quantify. This calls for the development of a new generation of UQ methods that go beyond parametric sensitivity to include model misspecification and for AI/ML methods such as physics-informed ML or operator learning, that are able to learn the most adequate functional form of the higher-scale models themselves. Another important source of uncertainty stems from the limited amount of sampling that can be carried out at each scale. At least in theory, the question of completeness can be addressed using known numerical analysis tools such as numerical continuation and deflation. These approaches remain mostly unused in the materials community and, at least in their standard form, come with a number of restrictions, e.g., the need for Hessian information, that should be addressed before their use can be generalized in production.

3. The need for concurrency will also mandate the use of scalable computational frameworks for workflow execution that make dynamic resource allocation possible in contexts where the computational workload varies strongly in time. These frameworks should also provide reproducibility and data provenance to ensure that the massive amount of information generated can easily be shared with the community. Furthermore, the heterogeneous nature of exascale computers requires the execution frameworks to be aware of the affinity between computational tasks and hardware units (e.g., certain codes can make efficient use of GPUs, while others can't). These challenges are discussed in Section 2.

Progress at IPAM

During the IPAM program, we took important steps towards addressing some of the outstanding challenges described above. The group identified the most pressing needs as the quantification of uncertainty as information is transferred between fidelity levels, and the use of this information to steer the allocation of computing resources, a currently overlooked aspect. During the program, we developed a novel methodology for UQ when moving from high-fidelity quantum calculations to low(er)-fidelity atomistic simulations which rely on machine-learning interatomic potentials to provide reliable confidence intervals on the predictions of the low-fidelity models. With this capability in place, we considered a motivating application to demonstrate how high-fidelity quantities of interest, in our case the diffusion tensor of defects in bulk materials, can be obtained at a much lower computational cost compared to a direct brute-force calculation at high-fidelity. In particular, the framework relies on low-fidelity atomistics to explore the local potential energy landscape of a defect in a bulk material. The low-fidelity model is ideal for this purpose as it is

comparatively inexpensive. However, inaccuracies in the state-to-state dynamics of defect transport calculated with the low fidelity atomistics, as characterized using UQ, can lead to exponentially inaccurate defect diffusion properties, which can be corrected by carrying out targeted high-fidelity simulations.

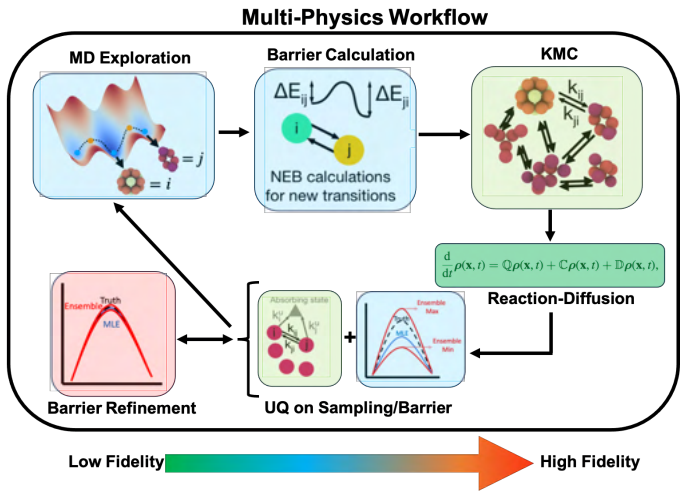

Fig 4: Schematic of the implemented multi-physics workflow that generates high-accuracy defect diffusion constants. Low-fidelity methods provide an abstract graph representation of state-to-state defect kinetics, which is upscaled into reaction-diffusion equations. Medium-fidelity methods involve a machine-learned interatomic potential to explore the energy landscape and provide an initial approximation of the energy barriers connecting states. Finally, a high-fidelity calculation methodology can provide refinement of the key calculations, increasing the accuracy of the quantity of interest, albeit at a very high computational cost. Crucially, the uncertainty quantification (UQ) considers (on an equal footing) the uncertainty derived from multiple fidelity levels, providing an on-the-fly coupling between these levels and enabling optimal resource allocation at scale.

The final ingredient was the development of a decision-making process to balance allocating computational resources between low and high-fidelity calculations. To do so, we developed a figure of merit that is an explicit function of the UQ metrics which can be optimized by carrying out either additional low or high-fidelity simulations. This approach makes it possible to optimally assign work to different fidelities so as to maximize the improvement in the figure of merit per unit computational cost. This method is expected to be inherently scalable and able to optimally allocate massive amounts of computing resources. A schematic representation of the prototype workflow is outlined in Figure 4 and seeks to generate a high accuracy defect diffusion constant from the reaction-diffusion equation by invoking a number of multi-physics approaches throughout the modeling spectrum of Figure 3.

The other aspect that the group decided to focus on was addressing the challenge of employing numerical continuation and deflation tools in atomistic simulations to identify critical point configurations, aiming at efficiently exploring the potential energy landscape in a way that could increase confidence on the completeness of higher-scale models. A proof-of-concept implementation was developed as a wrapper around `LAMMPS`, allowing us to trace continuous curves of critical points, including saddle points. Importantly, the

tool is entirely Hessian-free and is very general - it takes a few lines of code to adapt it to work with other systems and different continuation parameters. It has already proven to provide insight that is near impossible to capture with known dynamical approaches. Furthermore, we proposed a prototype deflation saddle point search technique, which acts to modify the potential energy landscape when conducting saddle point searching methods in order to bias the searching method away from previously discovered saddle points. The prototype, employed on a number of analytically defined surfaces, produces promising results and will now be applied to real atomistic simulations. Both of these methods are highly parallelizable and are envisioned to supplement known tools for exploring the potential energy landscape of atomistic systems, such as MD simulations coupled with nudged elastic band and dimer method calculations.

Vision statement

As discussed, the paradigm of multi-physics modeling is a tried and tested approach to tackling complex scientific problems. However, current approaches typically entail multidisciplinary collaborations that span many years, which drastically limit the ability of these methods to explore the chemical space of possible materials. The vision of the program participants is that exascale computing could enable a qualitative change in our ability to design better materials by decreasing the timescale for the prediction of performance for a new structural materials candidate from years to days. This could potentially have a transformative societal impact that could drive, e.g., the green economy revolution.

With the introduction of methods such as those developed during IPAM, the community can begin to consider integrated workflows which incorporate, in an online way, different levels of physics. The participants have identified research directions that should be prioritized over the next 5 years:

- The development of rigorous and ultra-scalable UQ methods for multiphysics modeling
- The development of high-performance frameworks that enable the deployment of complex computational workflows required for multi-physics applications
- The development of new physics-informed mathematical formalism for scale-bridging which would allow for the control of model-discrepancy errors

4. Co-Design for Exascale Architectures

Scientific opportunities

1. Fast and Accurate Atomistic Materials Simulations

Understanding materials behavior and designing novel materials requires efficient, accurate, large-scale, and long-time simulations of complex materials. Atomistic simulation methods, including molecular dynamics (MD) and structural relaxation, are an indispensable tool for computational experiments of materials behavior, enabling *in silico* discovery of new materials. Atomic forces calculated as a function of atomic positions are a critical input, and can be obtained

from quantum mechanical *ab initio* methods such as Density Functional Theory (DFT). However, *ab initio* methods are computationally expensive and have unfavorable scaling with system size, and are incapable of reaching the length and time scales required for many applications. Much progress has been made in the past decade in the development of machine-learning interatomic potentials (MLIPs), which are high-accuracy computationally efficient machine learning surrogates for *ab initio* forces. The cost of simulations with these MLIPs scales linearly with the number of atoms. Moreover, the high computational intensity gives excellent strong scaling on leadership computing platforms. This comes at the cost of design complexity, both in the MLIP training process and implementation in production software. The co-design of MLIPs and the resulting atomistic simulations for the exascale will enable quick development of models, and efficient deployment in large-scale simulations, enabling scientific discoveries in a variety of important applications.

2. Training Data Sampling

There exists a need to effectively sample training configurations to train interatomic potentials and bridge the gap between DFT and MLIPs. Domain-expert researchers typically possess significant prior knowledge that can inform ML potential training. For example, researchers may be interested in obtaining the lowest energy configuration of a system for property prediction or the minimum energy pathway between different states to estimate the transition rates for a kinetic Monte Carlo simulation. In these scenarios, it is of interest to sample training configurations from the relaxation pathway. Alternatively, many researchers desire to obtain potentials that are dynamically robust under certain atomistic simulation processes. This can include a wide variety of user-defined simulation protocols, such as successive and repeated sequences of relaxation pathways, melting, quenching, compression, that replicate physically and technologically important materials processing conditions.

3. Multiple Simulations on a Single GPU

The emerging exascale computing machines are composed of large numbers of heterogeneous hardware devices (GPUs), each containing a large number of concurrent processors allocated to a single runtime process. Possibilities to effectively utilize these resources for atomistic simulations are poorly understood. To provide sufficient parallel work to fully saturate all of the processing power of the device with a single simulation may require many millions of atoms. However, many useful atomistic simulation methods require relatively small atom counts with limited parallelism that will leave most of the GPU sitting idle. New hardware and software methods are needed to enable large numbers of independent simulations to run efficiently on a single device.

Current state-of-the-art

This IPAM program began with a rich foundation of state-of-the-art models and tools that were available for us to improve upon. Regarding our first challenge of fast and accurate atomistic materials simulations, we note that many ML potential models and fitting packages existed at the start of the program. The ML models supported by these packages did not provide easy portability

to new architectures. Furthermore, existing packages primarily supported a rigid workflow, such as simply fitting a model to a given data set of *ab initio* computed energies and forces. Such rigidity prevents the exploitation of active learning strategies that can sample training configurations dynamically. For example, there is a need to sample along minimization paths and during dynamical processes like compression/decompression. Since the aforementioned methods make use of DFT calculations for training, it is important to note current numerical methods in DFT lack robustness for complex material systems, such as heterogeneous and magnetic systems. Finally, regarding maximum use of exascale resources, current GPU acceleration abilities in `LAMMPS` and other packages do not support multiple simulations on a single GPU.

Progress at IPAM

An example of improvements in code portability achieved at IPAM during this program was the initial `LAMMPS/Kokkos` implementation of the Ultra-Fast Force Field (UF3). In addition, we established a protocol and obtained the initial results for a benchmark study looking at the trade-off of prediction accuracy and computational efficiency of several MLIPs. The initial benchmarks (Figure 5) show that UF3 gives the best performance for low to moderate accuracy while the quadratic Spectral Neighbor Analysis potential (QSNAP) gives the best performance at high accuracy. The simplicity of the UF3 model allowed the `LAMMPS/Kokkos` implementation to be completed in just a few weeks during the IPAM program.

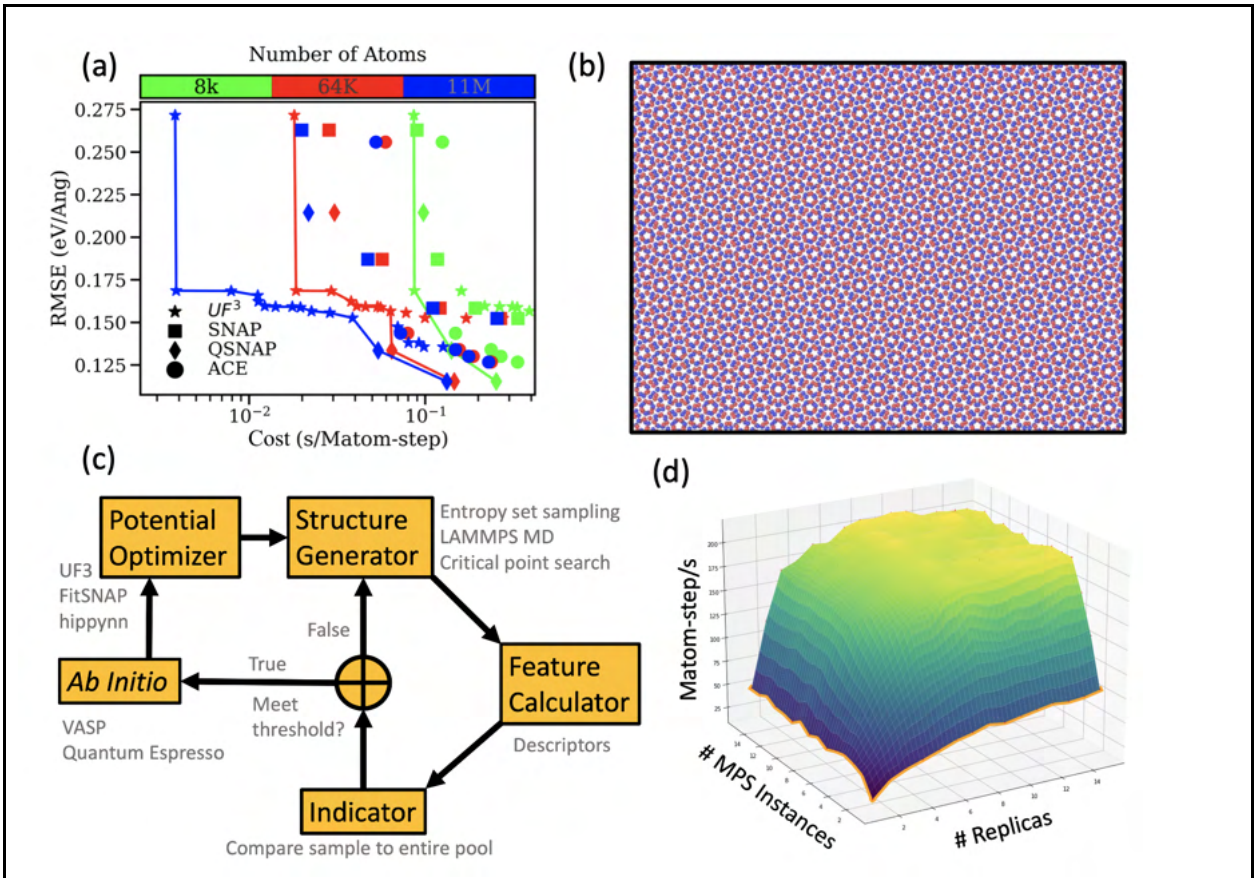

Figure 5: (a) The figure shows the force prediction errors on a hold-out test set for silicon as a function of computational cost. The markers represent the respective ML potential while the color indicates the system size i.e. number of atoms. The solid lines represent the Pareto front for the given system size. (b) Close-up of 2D MoS_2 twisted bi-layer quasicrystal approximant (θ=29.98864°) supercell with 45,378 atoms. (Anikeya Aditya, University of Southern California). (c) Our general active learning workflow with core abstractions that allow modularity and extensibility. The core abstractions are represented as blocks, with adjacent text highlighting examples that could be used for each module. (d) Surface of total efficiency (z-axis) for multiple simulations running on a single GPU device with combined over-subscription models (x and y axis).

Additional advances in electronic-structure predictions at the exascale were made. Our efforts at IPAM focused on using the Frontier exascale machine to study quasiperiodic systems that require large length-scales to accurately capture the relevant physics. Apart from the target application, the broader question we seek to answer is whether the length-scales and accuracy achievable by the state-of-art large-scale DFT codes with systematically convergent basis sets are sufficient for general studies of quasicrystal properties. In particular, we used the exascale ready `DFT-FE` code, a massively parallel adaptive finite-element based Kohn-Sham DFT code to compute the planar electrical and thermal conductivities of a twisted bi-layer MoS_2 system near the quasicrystal magic angle of θ=30°. While thermal conductivity can be adequately described by a classical interatomic potential, calculations of electronic conductivity require DFT. We have so far

performed structural relaxation and Kubo-Greenwood electrical conductivity calculations with 8000 atoms (70,000 electrons).

To tackle the challenge of efficiently sampling configurations for training MLIPs, we conceptualized the abstractions needed to implement a general active learning workflow, illustrated in Figure 5c. Our active learning framework starts with the structure generator module. This abstract module generates configurations of atoms, which may be achieved in different ways such as molecular dynamics, sampling from a large pool of entropy maximized structures, or sampling the relaxation trajectory of structures produced in saddle point searches. After the structure generator, the feature calculator extracts structural descriptors which are used by an indicator (UQ) to determine whether a newly seen structure is uncertain or not. If an uncertain threshold is met, an *ab initio* calculation is launched and the model is retrained. One example of this loop we explored is critical point searches, such as geometry optimizations and saddle searches. Preliminary results obtained from this long program suggests that a 5x reduction in DFT evaluations may be feasible. A refactoring of the `FitSNAP` library also allowed flexible active learning workflows like shown in Figure 5c.

In order to address the GPU undersaturation challenge, two different approaches for running multiple simulations on a single GPU were explored during the program. The first approach used the hardware specific Multi-Process Service (MPS) to assign multiple `LAMMPS` instances to a single GPU. The second approach uses a single `LAMMPS` script to initialize many independent replicas in the same simulation box. A modified `LAMMPS` neighbor list was invoked to turn off interactions between replicas. The `LAMMPS` source code was modified to improve efficiency. Both of these over-subscription methods improved GPU utilization and computational throughput to some extent. Furthermore, combining both methods yielded additional GPU utilization increases. For example, for a system containing 13,000 EAM atoms, we were able to run 13 `LAMMPS` MPS instances each with 13 independent replicas. The 169 simulations running on a single GPU achieved a total performance of 210 Matom-step/s, which translates to 5.5 μs of simulated time per day. Without these methods developed here at IPAM we would only be able to run a single replica per GPU, with roughly 100x less simulated time per GPU per day.

Outstanding challenges

On the active learning front, it is not surprising to note that a majority of the time is spent in DFT calculations. The general active learning workflow we described is sequential in nature; we check model uncertainty metrics every few timesteps, run DFT calculations, and fit to the results. The nature of this procedure requires waiting for a DFT calculation to complete before more progress on configurational sampling and training is made. To overcome this challenge, we require asynchronous active learning workflows whereby multiple active learning trajectories can be launched in parallel. This capability could be achieved with asynchronous methods described in the chapter on “Workflows for the Exascale”. Effective sampling of configurations can also be aided with descriptor signal processing tools described in the chapter on “Material Informatics at the Exascale”.

Robust convergence of DFT calculations is increasingly becoming a major challenge, particularly for material systems with complex features such as reactive chemistry, solid-liquid interfaces, extended defects, magnetic effects and net electric charge. Another emerging challenge in the exascale era is the increasing use of automated workflows generating large numbers of parallel DFT calculations. These would significantly benefit from robust and efficient numerical methods such as preconditioners to ensure reliable accuracy, especially for magnetic systems.

A continuing challenge is how to fully utilize hardware devices for simulations without arbitrarily increasing the system size or the arithmetic intensity of the MLIP. One solution explored at IPAM was running multiple simulations on a single GPU. An additional need on this front is better profiling to help users determine how effectively they utilize individual hardware devices. This would promote efficient use of resources when performing complex workflows, such as needed in a variety of applications and in active learning contexts.

Vision statement

The landscape of MLIPs and hardware will continue to evolve. In the case of hardware, we can expect greater complexity, but at the same time we will see the emergence of standardized programming models, such as support for `Kokkos` in the C++ standard. For MLIPs, we can expect growth in model complexity (e.g., equivariant descriptors, message-passing models, new physics). At the same time, carefully benchmarking will help identify compact models like UF3 that achieve expedient tradeoffs between accuracy and cost. We envision that the combination of all these factors will lead to greater choice for users of exascale software resources, enabling them to find the best combination of model and hardware for their science application. Alongside this ability to port new models to emerging architectures quickly, we envision a revolution in the way computational materials scientists utilize exascale resources. In the future, workflow managers will utilize vendor-specific tools to most efficiently pack many independent simulations on a single accelerator device. For example, it will be possible to run thousands of active learning trajectories on one device, allowing the most efficient dynamical sampling of training structures for fixed computing resources.

Our ultimate vision for the active learning workflow is that it matures into a general framework whereby users can obtain potentials for a variety of applications on exascale machines in under a day simply by letting the procedure run and make its own decisions. The vision of such a mature framework requires intelligent use of resources across the wide variety of computational tasks shown in Fig. 5c This includes mathematical methods and hardware-aware implementation advances in fast and large-scale *ab initio* calculations, allowing routine calculations of system sizes consisting of a few thousand atoms along with robust convergence for material systems with complex features. Further, we envision being able to conduct very large-scale DFT calculations reaching 50,000 atoms (500,000 electrons). This will allow validation of our trained potentials on much larger scales, such as with extended defects and solid-liquid interfaces. Our vision is that all of these tasks will be executed asynchronously to achieve maximal use of available resources. Such an advance will use new tools developed in “Workflows for the Exascale” and mathematical modeling strategies further discussed in “Materials Informatics at the Exascale”.

In summary, we have identified 3 outstanding challenges that warrant more investigation in the next few years:

- A systematic investigation and mathematical analysis of the efficiency/accuracy tradeoff in ML potentials for materials
- The development of new tools and algorithms that intra and inter-device parallelization
- The development of UQ-driven workflows at scale to systematize the parameterization and assessment of ML potentials for materials

5. Materials Informatics at the Exascale

With the first exascale computing systems now a reality, we are in an era where an unprecedented level of computational resources are available. Scientists are presented with new opportunities to conduct simulations at unprecedented scale and fidelity. On the other hand, this abundance of resources brings its own set of challenges: processing exabytes of data manually/offline to advance scientific understanding has become a bottleneck. Several months of data analysis followed by inferences to perform new simulations severely limits the possibility to exploit the full potential of exascale machines.

As such, complex software workflows to increase the automation are required, as discussed and introduced in Sections 2 and 3. In this section, we discuss the need for the testing, extension and development of new mathematical tools to inform the data handling for exascale materials science simulation as part of these workflows. We address the possibilities of developing mathematical models and efficient data analysis algorithms that can process the data generated in such a manner that it replaces low-level human decision-making, enabling effective utilization of resources. In addition, we also discuss the need to develop mathematical methods that give an effective way to store data. Addressing both of these challenges is essential to fully leverage the new generation of exascale machines.

Scientific opportunities

Over recent decades, a series of mathematical methods have been proposed within computational materials science to encode the key features of atomistic configurations in three-dimensional space. These approaches propose “descriptors" which integrate out physical symmetries, providing efficient bases to represent the environments around individual atomic nuclei. Examples of such descriptors discussed during the programme include the UF3 potential descriptors, Spectral Neighbour Analysis Potential (SNAP) descriptors, and the Atomic Cluster Expansion (ACE) descriptors. Currently, these descriptors are primarily used to fit machine learning interatomic potentials (MLIPs) to achieve forces with quantum theoretic accuracy but at a much lower cost. Each set of descriptors is governed by “hyperparameters”. Development of mathematical approximation theory for these descriptors is at an early stage, but over the next decade, will likely play an important role in guiding hyperparameter selection.

Alongside these recent domain-specific developments, there are long-standing approaches in mathematics and data science for the reduction of generic data vectors and time series. Some important examples include Principal Component Analysis (PCA); random projection or sketching methods; Tucker decomposition; diffusion maps and the Mori–Zwanzig formalism. Given the need for high-levels of communication in any implementation of these approaches, algorithms need adaptation to address new challenges in computational materials science at the exascale, and recent developments in this area were discussed during Workshops I and II of the program.

Outstanding challenges

We have identified a number of both scientific and technical challenges in the long program at IPAM including hyperparameter identification, automatic decision making, exascale workflow optimization, and applications of mathematical and statistical algorithms.

Firstly, as explained above, descriptors allow the development of more accurate machine-learned interatomic potentials. However, their usage in the context of post-processing and statistical inference and the choice of suitable hyperparameters is less well understood, but appears to be ripe for possible development. Calibration of suitable space of hyperparameters and selection of relevant algorithms for decision-making based on particular objectives is an open challenge, but one which may be promising to help further automate exascale workflows.

A second challenge is the need to develop mathematical techniques for data storage that ensure efficient memory usage while enabling fast data retrieval. One specific application is the search for chemical structures within a simulation-generated database in order to expedite the simulation. For instance, the bottleneck of adaptive kinetic Monte Carlo (AKMC) simulations is the calculation of saddle points. Thus, previously calculated saddle points are stored in a kinetic database (KDB) along with corresponding atomic positions of reactant state and are used during simulation. During AKMC simulations, the code queries the KDB for similar atomic environments to find the saddle points on-the-fly by utilizing a neighbor list matching process. However, this can become very time-consuming and computationally-expensive when the database increases in size. During IPAM we showed that atomic environment descriptors can be used in changepoint detection (c.f. Progress at IPAM) which is a similar problem in nature to using atomic environment descriptors to describe and search for a chemical configuration in a database. Therefore, using atomic environment descriptors to store chemical structures and query processes based on them to find on-the-fly reaction paths and kinetics is a promising approach.

From the perspective of scale-bridging, i.e. using atomistics to inform higher order mesoscale models (e.g. crystal plasticity), exascale architectures can serve as ideal platforms to perform atomistic simulations on the order of billions of atoms and beyond, hence making massively parallel atomistic simulations routine. However, effectively automating the decision-making process for both the generation and post-processing of long and large-scale trajectories of data is an open problem. Developing new exascale-compatible materials informatics is therefore

crucial to enable on-the-fly multi-scale simulations, leveraging the full potential of exascale platforms.

Turning to data-reduction and representation, the development of statistical learning methods usually relies on probabilistic assumptions of the dynamics and parameters, which are rather difficult to justify in reality. Careful method selection is therefore required to apply existing statistical learning methods to material science data in the exascale. Moreover, traditional statistical learning methods are not suitable for truly massive data sets. Existing algorithms and approaches must be adapted to be efficient for parallel implementation on exascale machines.

Finally, to ensure workflow efficiency, automated decision-making methods must be developed to allow exascale allocations to be used effectively without significant human intervention. A first step towards designing automated approaches requires methods such as online event detection, so that only the data points with the highest information content are identified in the extremely high-dimensional space of atomic trajectories. In the longer term, the development of mathematical models to inform workflow management using approaches such as queuing theory and Bayesian optimization will be crucial. The massive availability of computational resources on exascale machines is a new paradigm for these theories, and accordingly requires the development of novel mathematical theory.

Progress at IPAM

During this IPAM long program, we explored the possibility of reducing trajectory data generated from MD simulations to only events of interest, allowing us to manipulate data during simulations, and hence minimizing communication overheads which are critical bottlenecks for exascale architectures. In particular, we focused on exploring methods to remove the need for human input to identify events as a natural first step towards the automation of decision-making within a workflow. Concurrently, we analyzed existing atomic descriptors with the aim of data reduction: we discuss each of these briefly below.

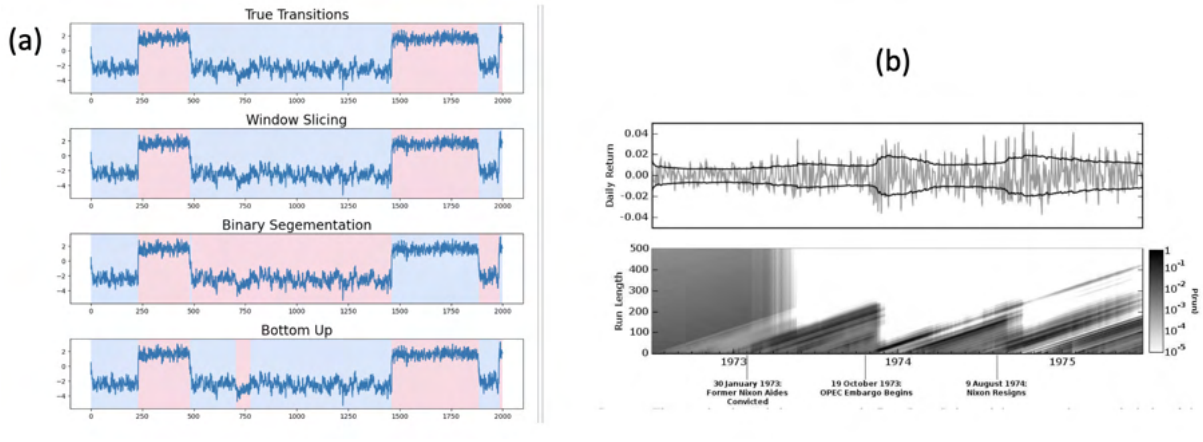

Figure 6: (a) Offline changepoint detection algorithms from `ruptures` package. (b) Posterior

predictive probability-based Bayesian online change point detection.

It is desired to replace human decision-making by algorithm-based decision-making. In particular, we look at labeling of when and where events of interest occur while the simulation is running. Using both synthetic data and real MD simulation data, we explored different “change-point” detection algorithms in Jupyter Notebooks. The algorithms use signal processing techniques to detect major changes in data, and can be broadly classified as online or offline. Offline in this context means that the whole time series needs to be known to detect a change point. Online methods on the other hand receive the data point by point as it is generated and can detect changes in this continuous stream based on recursive posterior estimates. Alongside this, we explored the impact of both varying descriptor hyperparameters combined with the PCA algorithm as a way to efficiently reduce data while still identifying events. Many of the leading descriptors used in computational materials science hyperparameters governing the level of nonlocality in space, the level of regularity, and the order of approximation. By exploring this hyperparameter space, we began developing a new understanding of the robustness of these descriptors for information compression, and this study will continue after the program has concluded, extending the community’s understanding of these powerful new tools.

Vision Statement

The achievement of these goals requires a concerted cross-disciplinary effort, with communication between software developers, domain experts, mathematicians and data scientists, and the highest standards of openness and collaborative sharing of best-practice. We believe that success in these aims will enable us to accelerate advances in our understanding of problems in materials science by deepening our understanding of the underlying mathematics and materials science, allowing more efficient data sharing and re-use, and ultimately optimizing the use of both human and machine resources.

As a research community we believe the most pressing challenges which must be addressed are the development of:

- New techniques to effectively compress and store high-value data generated from exascale simulations; and
- Robust mathematical models and techniques which enable us to inform low-level decision-making for complex scientific workflows.

6. Making the Exascale Accessible for its Expected Users

Computing in general, but even more so exascale computing, entails some challenges for the individuals that operate it. Solving at least two problems is essential to enable effective use of the vast power that exascale platforms offer:

(1) User interface: Can the user interface to exascale computing be designed in a way to allow fast and efficient access to the scientific community at large, especially to those without a sophisticated computational background?
(2) Knowledge extraction: How to ensure that the essential knowledge from the vast stack of information generated by an exascale simulation actually reaches the human who requested it?

Both questions are critical, considering the sheer cost of an exascale simulation. For instance, the US Department of Energy's first exascale computer Frontier was built at a $600 million cost. Thus, assuming a lifetime of 5 years (i.e., around 43,800 hours), one hour of simulation time on the full machine costs $13,700, excluding costs for power, operating staff, etc. This is not a low amount. Ensuring that this time is used wisely is important (even given the context of "research", which, by necessity, must allow for failed experiments).

The challenge is that the users in question may be domain scientists, for whose applications the computer was actually purchased, and often students. Computer systems need to be manageable by these users in order to maximize productivity achieved by these powerful resources. Workforce development and training of these users on the exascale systems in question, as well as a focus on ease of use of the machines, are critical to maximize the value of the exascale investment. For the ideal users of the exascale, deep domain knowledge and simultaneous expertise in advanced computer technologies must go hand in hand.

Current state-of-the-art

Figure 7 shows a schematic illustration of a portion of an exascale computer - typically, thousands of individual compute nodes, connected by fast communication networks. At the hardware level, each node will contain multicore CPUs (around 100 cores per node are typical), working memory (actually, levels of memory of different latency), as well as accelerators, commonly referred to as GPUs. This hardware layer is, in principle, transparent, but the complexity of the software ecosystem associated with each different machine can pose major challenges for end users. Libraries that are typically visible to user codes include message passing interface (MPI) libraries and shared libraries that perform system-specific, hardware-optimized tasks such as linear algebra and other mathematical tasks. These libraries are linked to user codes via compilers that support CPU execution, GPU execution, or both.

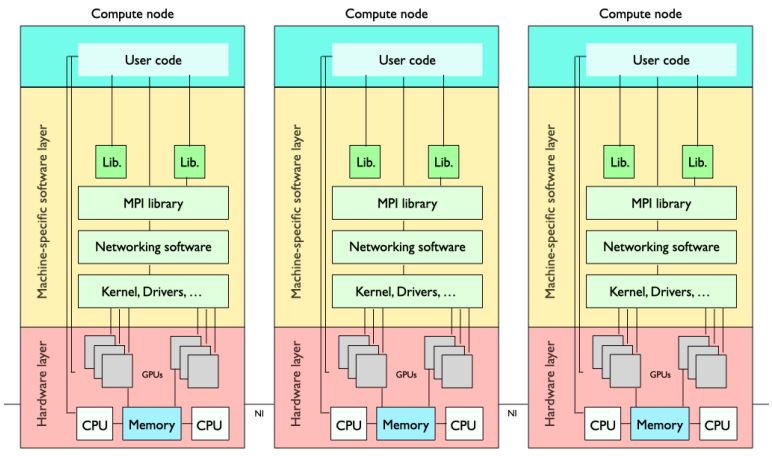

Figure 7: Schematic image of present-day exascale architectures. Multicore CPUs and multiple GPUs are integrated in individual compute nodes, connected by fast network interfaces (NI). User codes on these machines build on machine-specific shared libraries, which will vary in implementation and versions from machine to machine. For cross-node communication, the message passing interface (MPI) protocol is typically essential, however, this technology relies itself on further layers of software and drivers. Overall, a combinatorial explosion of moving pieces results that are rarely included in full when training expected users.

The point of Fig. 7 is that these are not simple systems for any user. Unfortunately, the different possible versions of each component (hardware or software) are ubiquitous, as are different vendor specific abstraction layers and standards that hide different pieces from their users. Incompatible software layers pose major problems for users who are frequently faced with a combinatorial explosion of possible environment settings. These are issues that were known challenges for the Exascale already at the outset (in the 2000s) but they nonetheless remain pertinent issues.

Science and even engineering students in the U.S. frequently are rarely exposed to computational education beyond simple scripting languages, e.g., python, i.e., the underpinnings of modern high-performance computing are largely not known to university graduates (arguably, already at a rather high and specialized level of education) in the STEM disciplines. Yet, the population that uses HPC resources is recruited from this group.

Outstanding challenges

User Interface and User Training - the Input Side of the Exascale

How do we train individuals or integrated research teams interested in science (i.e., in the results of computation, rather than in the science of the computer itself) to understand the machine - a research tool of immense power - sufficiently well to operate it meaningfully (i.e. to provide the right input information so the machine is doing exactly what the user is wanting and expecting it to do)?

This is not a trivial question, since scientists are usually, and for good reason, already busy delving into complex scientific topics in their own domains, rather than, e.g., the latest innovations of the networking layer underlying their MPI library. Simply mapping the full complexity of the computational resources, which reflects the state of the art at many centers today, cannot be the only pathway forward.

Interpreting Data - the Output Side of the Exascale

Even assuming that the complexity of the Exascale is mastered by a user: how do we train humans to learn and understand the fairly complex scientific concepts underlying the powerful workflows that they now have at their disposal?

Eugene Wigner famously remarked "*It is nice to know that the computer understands the problem. But I would like to understand it too.*" This is certainly true for exascale simulations that generate a tremendous amount of data. In a simulation of plasticity, for instance, we may see how a material deforms from an initial to a final state - however, pinpointing exactly why and where a material failed (and maybe how to prevent that failure) is buried somewhere in the data. Understanding this still requires a human to interpret and sift through an overwhelming amount of data, ideally with the help of state-of-the-art tools such as those discussed in Sections 5.

To avoid major data misinterpretation, good practice should include an informed forecast of the expected results - at least the reasonable computational order of the results should be known to the user. This sounds trivial but is really also a matter of good workflow design, minimizing complexity while keeping the essential pieces, and training. To be able to do so, it is essential to understand the science behind the simulation.

Understanding the essential science behind a simulation

We may increasingly find ourselves in the situation where the human operating the computer may not themselves have the training to understand the science behind the simulation that they just ran. Our workflows tend to embody increasingly sophisticated, complex and realistic theories that

may themselves require several college-level courses and a few textbooks beyond the college level to understand fully. How can we guide real-world users, often advanced students, to ensure that the theory is understood and trivial but costly mistakes are avoided?

It is certainly true that knowing every last detail of a simulation is not necessarily desirable or even feasible when the objective is to extract meaningful scientific output from the result of 10^{18} mathematical operations per second. For example, few in materials science need to know how a spherical harmonic function is computed, as long as the computer does that math correctly. However, things are different when the scientific formalism that was employed itself runs into its limits for a given task and gives a physically wrong result. Running a simulation of a solid that cannot, for instance, deform plastically because the underlying approximation to the potential energy surface was limited is perhaps possible, but generally not helpful to understand the behavior of a real material.

Progress at IPAM

Much of the above problems are not new, but they nevertheless remain open challenges. The work summarized in this document reflects some preliminary solution to the challenge that is the exascale: Creation of infrastructures that facilitate effective code design, creation of software tools that enable more efficient extraction of the critical science insights from the vast amounts of data generated by an exascale simulation, creation of workflows - via jupyter notebooks - that abstract and centralize the most complex machine specific decision making processes and allow a user to follow their project tasks step by step in a visual way, while retaining full control of their user segment of the computer.

A key opportunity from those workflows, though rarely realized, would be to embed scientific training into the workflows themselves - that is, ensure that critical science steps are explained alongside a workflow, for users who are not already familiar with the background. Much of the science embodied in the workflows of interest in materials science requires one or more graduate level courses, e.g., in physics. In a workflow, the execution steps of those scientific tasks are already separated from their derivation/explanation, encouraging the frequently catastrophic habit of black-box usage of what may be hundreds of thousands of dollars worth of computing resources. Working towards guided workflows that convey essential understanding to a user, along with the results, could address this.

Vision Statement

It should be acknowledged that the assembly, operation and maintenance of pre-exascale and exascale machines is itself a major challenge. While such challenges are largely outside the scope of this document, as users of HPC environments, we are impacted by downtime or other issues that may come up as these new machines come online. Perhaps one takeaway from dealing with such issues, is that clear lines of communication need to be available between those

standing-up pre-exascale and exascale facilities so that users can both understand what challenges exist on the hardware side, while also making clear their needs and desires.

The user interface side, however, i.e., teaching all users the basics of the computers they are about to use and how to maximize sensible productivity on these machines will remain perhaps the number one challenge to be solved, in order to realize the full potential of this technology.

- Simplicity, without oversimplification, must be a guiding principle of exascale architectural design. Transparent user interfaces without pitfalls matter, even if the basic interface is the command line. Software layers that hide essential decisions pose risks the heart of an exascale computer lies the hardware layer and the details of that hardware layer must be accessible and transparent to a user.
- It follows that teaching the basics of computer architectures and how their components interact should be part of STEM education, and in principle, it might be helpful if any user of a HPC system, particularly exascale, were asked to be trained on the basic principles of the machine before being granted access. While this creates a bureaucratic obstacle, the reality is that training is currently optional or outsourced to the grass-roots base of science users of exascale type machines (whose scientific output justifies the existence of these machines in the first place). Ensuring sound, basic training of all users can go a long way to maximize the benefit of exascale machines.
- Can we use workflows to teach the essential pieces science of a given problem along with the user executing the workflow? E.g., can we incorporate links that explain the essential equations, their derivation, approximations made and their limits to a user as the workflow progresses along? This might greatly lower the entry barrier for new users and newcomers to exascale computing.

Appendix: A Non-Human Perspective

Lazy Human: What are the challenges and opportunities for computational materials science at the exascale?

ChatGPT: Computational materials science at the exascale presents both challenges and opportunities. Here are some of the key challenges and opportunities in this field:

Challenges:

1. Scalability: Achieving efficient scalability of computational models and algorithms to effectively utilize the massive parallelism of exascale systems is a significant challenge. Designing algorithms that can efficiently distribute workloads and manage data across a large number of processing units is crucial.
2. Heterogeneous architectures: Exascale systems often employ diverse processing units, such as CPUs, GPUs, and specialized accelerators. Exploiting the unique capabilities of each hardware component and developing algorithms that can effectively utilize these architectures require a deep understanding of their programming models and optimizations.
3. Memory and I/O bottlenecks: The increased computational power at the exascale exacerbates memory and input/output (I/O) bottlenecks. Co-designing strategies to minimize data movement, optimize data locality, and utilize high-bandwidth storage solutions are crucial for reducing these bottlenecks and maximizing performance.
4. Algorithmic adaptability: Traditional algorithms in materials science may not be well-suited for exascale architectures. Co-designing adaptive algorithms that can dynamically adjust their strategies based on the available computational resources is essential for efficient utilization of the hardware.
5. Software infrastructure: Developing robust and efficient software infrastructure for exascale materials science simulations is a challenge. This includes frameworks for high-performance computing, data management, visualization, and analysis, which need to be scalable and capable of handling the massive amounts of data generated at the exascale.

Opportunities:

1. Enhanced accuracy and complexity: Exascale computing enables simulations with increased accuracy and complexity. With greater computational resources, researchers can incorporate finer details into their models, such as atomistic simulations or quantum effects, leading to more realistic and accurate predictions of material behavior and properties.
2. Multiscale simulations: Exascale computing facilitates multiscale simulations, where different length and time scales can be seamlessly integrated. This enables researchers to investigate material behavior across multiple levels, bridging the gap between atomistic simulations and macroscopic properties, and providing a deeper understanding of materials phenomena.
3. Materials discovery and design: Exascale computing can accelerate materials discovery and design processes. By performing large-scale simulations, exploring vast chemical

spaces, and leveraging data-driven approaches, researchers can identify novel materials with desired properties, reducing the time and cost involved in experimental trial-and-error.

4. Data-driven approaches: The exascale era generates massive amounts of data. Co-designing techniques that combine simulation and data analytics, machine learning, and artificial intelligence can enable the extraction of valuable information from these large datasets. Data-driven models can complement traditional simulation-based approaches, leading to new insights and accelerating materials research.
5. Collaborative research: Exascale computational materials science fosters collaboration among researchers from different disciplines, such as materials science, computational science, mathematics, and computer architecture. Collaborative efforts enable the exchange of expertise, the development of novel methodologies, and the acceleration of scientific discoveries.

In summary, computational materials science at the exascale offers opportunities for enhanced accuracy, multiscale simulations, materials discovery and design, data-driven approaches, and collaborative research. However, it also poses challenges related to scalability, heterogeneous architectures, memory, algorithm design, and software infrastructure. Addressing these challenges and leveraging the opportunities can revolutionize materials research and facilitate the development of new materials with tailored properties.