Review

Designing physics experiments with artificial intelligence

https://doi.org/10.1038/s41586-026-10898-6

Received: 15 June 2025

Accepted: 9 July 2026

Published online: 2 September 2026

Check for updates

Jonathan Klimesch[1]✉, Sören Arlt[1,2], Carlos Ruiz-Gonzalez[2,3], Carla Rodríguez[2], Xuemei Gu[2,4], Philipp Haslinger[5,6], Pietro Vischia[7], Christian Haack[8], Yehonathan Drori[9], Rana Adhikari[10], Markus Arndt[11], Michael Kagan[12], Lukas Heinrich[13] & Mario Krenn[1]✉

Progress in physics has long been driven by ingenious experiments conceived by human experts. Recently, design methods driven by artificial intelligence (AI) have begun to move beyond tuning a handful of parameters to proposing entirely new experimental layouts. The discovered configurations often challenge established design conventions while matching or even exceeding the performance of human-designed set-ups. We frame experimental design as a search for optima over a vast space of hardware configurations subject to practical constraints and organize this Review around four guiding questions: how can we (1) engineer expressive search spaces; (2) build fast and reliable simulators; (3) translate scientific goals into computable objective functions; and (4) develop AI-based exploration methods that can navigate both discrete and continuous design choices. These questions place AI-driven design on a scale from parameter tuning to de novo discovery, and highlight the trade-offs between computational tractability, experimental feasibility, interpretability and solution reliability. Looking ahead, simulators spanning several physics domains, combined with large suites of experimental objectives, could discover unorthodox experimental concepts that are difficult to arrive at with human intuition alone. Ultimately, AI-designed experiments might thereby open new ways to explore the Universe.

Physics experiments are our way to ask questions to the Universe[1,2]. Historically, experiments have been designed by creative and experienced human researchers[3]. Yet, given the vast space of all possible experiments, it is unclear whether human intuition and experience alone are the best ways to discover new experimental techniques. Recent developments indicate an exciting possibility: intelligent algorithms are beginning to extend beyond optimizing small parameter sets towards fully automated, de novo experimental design. This emerging approach has already shown promising results across various domains of physics. In quantum physics[4–8], entirely computer-generated experiments have been successfully implemented[9–11] and led to new physical insights[12–14]. In plasma physics, this approach results in proposals of optimized fusion reactors[15,16] and in particle and nuclear physics, it contributes to the design of high-energy detectors[17–19]. In neutrino and astrophysics, it leads to designs of more efficient high-energy antennas[20] and improved detector arrays[21,22]. In gravitational wave physics, algorithms produce detector blueprints with improved sensitivity that are challenging for humans to interpret[23] and in optical microscopy, computationally discovered measurement methods could influence biological research[24,25].

On a high level, this design model is a search for optima of a complicated, multivariate function. The function output describes the quality of an experiment—for instance, its sensitivity. The variables are experimental parameters, for example, laser power or mirror reflectivity. Usually, it is neither necessary nor possible to find the highest-quality experiment as the global optimum: discovering configurations that outperform the best-known experiments is often already scientifically valuable—and a much simpler task.

Here we explain how and where this approach works at present, provide insights on future challenges and limitations and finally share our vision on how de novo discovery—the most extreme form of this approach—could lead to the discovery of new experiments that form an AI-generated test bed for fundamental physics theories. We centre our Review around four questions (Fig. 1):

- Search spaces. How can we define a large and complex search space that contains exceptional and surprising experiments?
- Physics simulators. How can we build fast and robust simulators or parametric models that sufficiently approximate experimental outcomes across the entire search space?

[1]Department for Computer Science, Faculty of Science, University of Tübingen, Tübingen, Germany. [2]Max Planck Institute for the Science of Light, Erlangen, Germany. [3]ELLIS Unit Linz, Institute for Machine Learning, Johannes Kepler University Linz, Linz, Austria. [4]Institute of Condensed Matter Theory and Optics, Friedrich Schiller University Jena, Jena, Germany. [5]Vienna Center for Quantum Science and Technology, Atominstitut, TU Wien, Vienna, Austria. [6]University Service Centre for Transmission Electron Microscopy, TU Wien, Vienna, Austria. [7]Departamento de Física and ICTEA, Universidad de Oviedo, Oviedo, Spain. [8]Department of Physics, FAU Erlangen-Nürnberg, Erlangen, Germany. [9]Raymond and Beverly Sackler School of Physics and Astronomy, Tel Aviv University, Tel Aviv, Israel. [10]Institute for Quantum Information and Matter, California Institute of Technology, Pasadena, CA, USA. [11]Vienna Center for Quantum Science and Technology (VCQ), Faculty of Physics, University of Vienna, Vienna, Austria. [12]SLAC National Accelerator Laboratory, Menlo Park, CA, USA. [13]Department of Physics, Technical University of Munich, Munich, Germany. ✉e-mail: jonathan.klimesch@uni-tuebingen.de; mario.krenn@uni-tuebingen.de

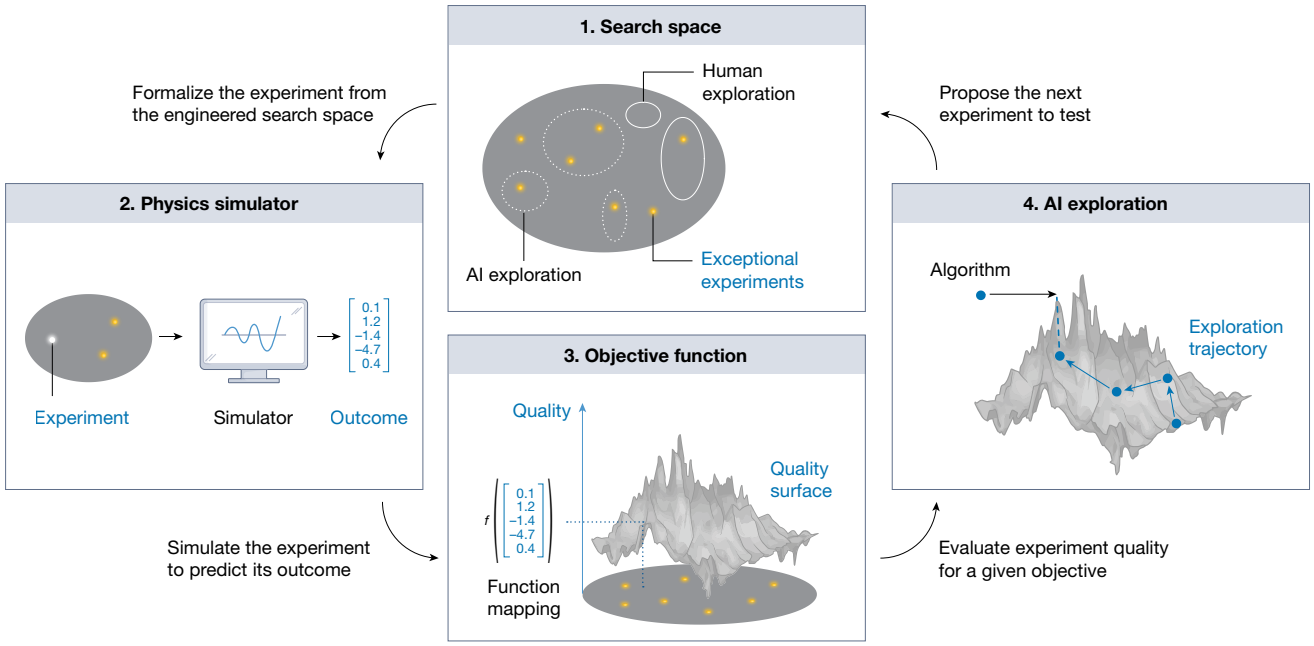


Fig. 1 | The four frontiers in physics experiment design with AI. Blue indicates inputs and outputs of each step. (1) Experiment candidates are selected from a carefully engineered search space that may accommodate exceptional designs. Some of these experiments may be discovered through human exploration, others lie outside human experience and intuition and may require complementary exploration techniques based on AI. (2) A physics simulator evaluates experiment candidates from the search space by computing their experimental outcomes. (3) An objective function, aligned with the scientific goal, converts simulator outputs into a quantitative quality score, forming a quality surface over the search space. (4) AI exploration algorithms use the entire exploration trajectory to propose new, higher-quality designs, creating an iterative improvement cycle.

- Objective functions. How can we engineer objective functions that describe desired and practical goals and constraints while being general enough to admit surprising solutions?
- AI exploration. How can we develop algorithms that explore the enormous search space and find solutions that outperform human-designed ones?

These four frontiers place any project on a scale from parameter tuning to de novo discovery. At one end, the search space consists of a fixed experimental topology with only a few optimizable parameters. At the other end, the search space contains topological choices and scales to thousands of discrete or continuous parameters.

AI-driven de novo design is already established in other domains and has yielded notable advances—in biology through de novo protein design[26], chemistry through the design of functional materials[27,28] and engineering through topology optimization[29]. Although physics experiment design often involves material design and topology optimization, it poses its own distinct challenges: reasoning over discrete experimental components, their physical interactions and parameters and diverse representations of experimental topologies.

With experiment design, we refer to the layout of experiment hardware. The design of operational procedures[30,31], control systems[32–34], measurement strategies such as Bayesian experiment design[35] or readout and data analysis methods[36,37] are discussed only when relevant.

Search spaces

The space of all possible physics experiments is enormous. A set of ten interferometry components can yield more than a billion possible topologies[23]. Even small spaces can already host all human-designed experiments within a given field. For example, a circuit with only three components (inductors, Josephson junctions or capacitors) includes all superconducting qubit designs reported in the literature[38].

To make such search spaces accessible for algorithmic design, researchers must first select subspaces that match their computational resources and scientific goals. This selection can be thought of as a multistep process that maps the search space to a range of subspaces—from the smallest, hosting a single component with only one parameter, to the most general de novo space of all possible physics experiments (Fig. 2). This process determines the physical scale of the experiment. A space with only small optical elements may yield tabletop experiments[4,5,8,25,39], whereas a space incorporating kilometre-scale vacuum tubes, large calorimeters or arrays of photomultiplier modules yields big-science projects[21,23,40]. A useful sanity check for successful search-space engineering is whether it can represent all known experimental solutions.

We discuss the search space engineering in three steps, illustrated in Fig. 2: the selection of components, topologies and parameters.

Experimental components

The first step is the selection of component classes that the search space should host. A useful distinction is whether selected components interact with each other. In quantum optics, beam splitters, mirrors and squeezers transform a quantum state. Combining such transformations yields a rich space of component interactions and can lead to the discovery of new quantum measurement schemes[6,12]. Gravitational wave detectors use similar components but at larger scales: mirrors and beam splitters interact through the propagating laser field, producing complex behaviour such as optomechanics that algorithmic design can exploit[23].

Other components, such as radio telescope antennas[41] or optical modules from neutrino detectors[21], do not interact but independently contribute to the final outcome. Here novelty comes from combining high-dimensional geometries with engineering or scientific constraints. Optimizing the arrangement of many independent elements can still yield high-quality experiments that deviate from human design principles, for example, producing ring-like layouts in which humans were focused on dense central cores[22].

The potential for surprising, high-quality solutions is therefore influenced by the number and diversity of component types and their interactions. However, allowing more component classes also increases

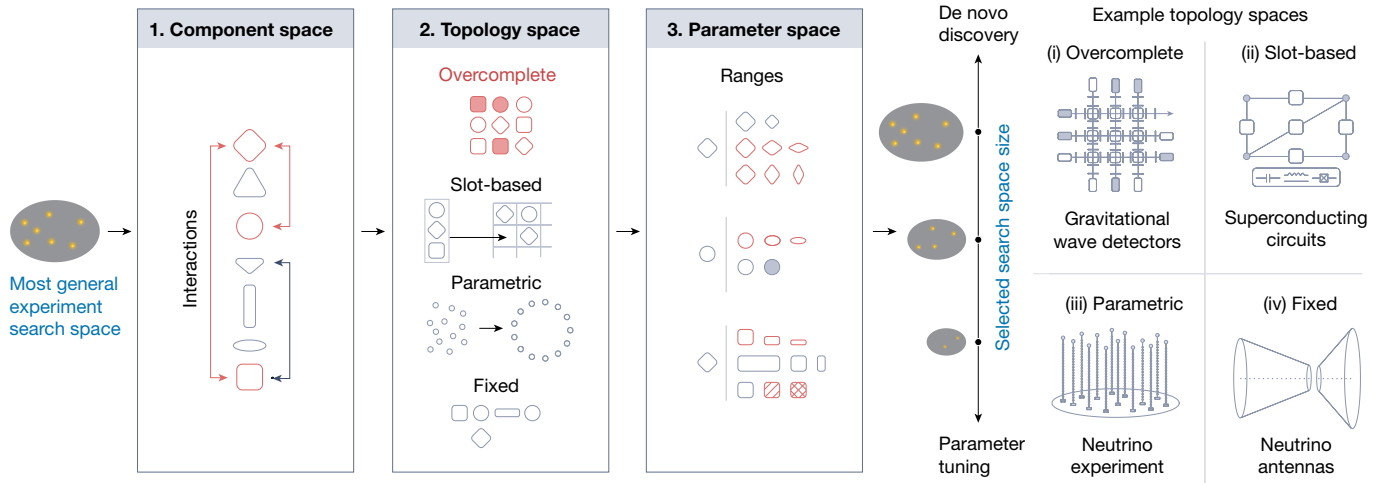


Fig. 2 | The search space engineering in three steps. From the most general experiment search space, researchers select a subspace that matches their computational resources and scientific goals. Red indicates an example selection and blue indicates inputs and outputs of the process. (1) Component space: the set of allowed experimental building blocks is chosen (for example, lasers, mirrors and detectors). Some components can interact (for example, by means of optical beams), whereas others remain independent. (2) Topology space: the allowed ways components can be arranged are specified. An overcomplete topology space uses a single, highly expressive ansatz that purposefully contains more components than needed and from which many effective subtopologies can be distilled through continuous exploration. A slot-based topology space allows iterative addition or removal of components at predefined locations, such as grid sites or graph nodes. A parametric topology space accommodates topologies that can change through continuous parameters. A fixed topology does not allow any changes to the component arrangement. (3) Parameter space: each component is assigned tunable parameters and allowed ranges (for example, a mirror mass between 10 g and 200 kg). Depending on the size of the selected search space, these design choices define a range from narrow parameter-tuning problems to broad de novo discovery spaces capable of yielding new experimental configurations. Example topology spaces are shown in (i)–(iv). An overcomplete gravitational wave detector space[23] (i), slot-based superconducting circuits[51] (ii), a parametric neutrino experiment space[21] (iii) and a fixed neutrino antenna topology[20] (iv).

simulator complexity, complicates exploration and risks producing solutions that are impractical to build.

Experimental topologies

The second step is to select a topology space, which defines the possible ways components can be arranged. We consider fixed, parametric, slot-based and overcomplete spaces, illustrated in Fig. 2. These are not static categories but mental models that can serve as guidance.

In a fixed topology, components and their arrangement can not be changed, reducing the search to pure parameter exploration. Neutrino antennas (Fig. 2) could be defined as an asymmetric bicone[20], limiting the exploration to cone lengths, radii and angles. A fixed topology is often used in proof-of-principle designs or detectors with strong engineering constraints[17,18,40,42–44].

In a parametric topology space, the topology can change through continuous parameters—for instance, by moving components in space. For neutrino detectors such as the Pacific Ocean Neutrino Experiment, or P-ONE (Fig. 2), the topology changes by optimizing the positions of 70 vertical one-kilometre underwater cables with optical modules[21]. Similar optimizations appear for seismometer layouts at gravitational wave detectors[45,46], arrays of radio telescopes[41] and detection unit placements for gamma-ray observatories[22]. Parametric topologies can also be expressed in a functional form, in which parameters control spatially varying properties such as material density[47], permittivity and refractive indices[48–50], or the geometry of stellarator coils[15].

In a slot-based topology space, the topology is changed by populating discrete slots with components. In superconducting quantum circuits, slots are empty spaces within a fixed circuit layout in which an algorithm can place electrical components[38,51] (Fig. 2). In quantum or high-energy physics, slots are placeholders for optical components or sensors, for example, organized in sequences[4,5,52], tensors[7] or graphs[39]. Slot-based spaces enable dynamic topologies that can grow or shrink in size.

An overcomplete topology space starts from one highly expressive ansatz—a topology in which many experimental solutions can be discovered by continuous optimization. Instead of building an experiment from scratch, solutions are distilled from one overcomplete template that purposefully contains many more components than necessary. One example of an ansatz for gravitational wave detectors is an interferometer grid (Fig. 2) designed to accommodate a wide range of known detectors, such as LIGO Voyager, and in which algorithms can discover unconventional designs with superior performance[23] (Box 1). Analogous ideas appear in super-resolution microscopy[25], quantum optics[6,53] and inverse design of nanophotonic components[48–50]. Inspired by neural architecture search[54], an ansatz can be formulated as an overcomplete probabilistic template in which all candidate components are initially included with learnable continuous weights. Optimization gradually suppresses irrelevant components and a concrete experimental design is recovered by retaining only those with the highest weights[55]. Constructing an ansatz is challenging and often requires sophisticated metrics of expressibility, that is, how many other experimental solutions the ansatz contains[56].

The choice of the topology space largely determines the type of exploration and the trade-off between expressivity and experiment complexity. Fixed and parametric spaces avoid discrete topology exploration. Fixed spaces have low expressivity but ensure experiment feasibility, whereas the expressivity and design complexity in parametric spaces are determined by the number of topology-defining parameters. Slot-based spaces require exploration of what to place where. Increasing the number of slots improves expressivity but also raises complexity. Overcomplete topology spaces are highly expressive at the cost of increased experiment complexity and the need for subsequent simplification. They also shift the type of exploration from discrete topology decisions to navigating large continuous parameter spaces.

Experimental parameters

After selecting component types and topologies, the third step is determining which parameters should be optimized over what ranges. One strategy is to fix parameters at predefined values. This yields a less

Box 1

Example for designing gravitational wave detectors with AI

Scientific goal: discover new concepts for detecting distant gravitational wave signals.

Search space

Component space. Component choices are often influenced by simulator capabilities. For gravitational wave detectors, Finesse[148] is a widely used, well-validated interferometer simulator that covers standard interferometric components, including lasers, squeezers, beam splitters, mirrors, suspensions and photodiodes. Because this set supports rich quantum and optomechanical interactions, we choose it as the component space.

Topology space. Known detector topologies are already optimized, so discoveries beyond incremental improvements are expected mainly from new layouts. A fixed topology space would mostly support engineering constraints or parameter tuning, whereas a parametric space is impractical because discrete optical components do not naturally map to continuous parameters. A slot-based space, in which components are added iteratively, is possible but would require complex exploration over both discrete and continuous choices and frequent recompilation of the potentially just-in-time-compiled simulator. We therefore choose an overcomplete topology space that relaxes the discrete–continuous search problem into a continuous one. We use an expressive rectangular grid ansatz[23], which can encode many known interferometer topologies while allowing potentially new ones to emerge (Fig. 2).

Parameter space. The optimizable parameters determine the simulation detail level. Mirror curvature, for example, is meaningful only in a Gaussian beam model, which increases computational cost. For simplicity, we use plane-wave-compatible parameters such as laser power, mirror reflectivity, mass and position. Parameter ranges are based on feasible existing designs: optical lengths, for example, extend up to 4,000 m, comparable with present gravitational wave detectors[149] and leading to big-science experiments.

Physics simulator

Finesse is fast for simulations but does not support efficient gradient calculation[23]. Because the underlying physics is deterministic, the simulator is highly suitable for automatic differentiation. However, efficient code augmentation is limited by Finesse's Python implementation. We rewrite the required simulator components from scratch in JAX[72]. The resulting simulator, Differometor, provides automatic differentiation, hardware acceleration, vectorization and just-in-time compilation. It matches Finesse in accuracy while enabling orders-of-magnitude faster gradient computations[77].

Objective function

We translate the scientific goal into a computable objective by maximizing detector sensitivity over a selected frequency range. The primary target-matching objective is defined as the summed logarithmic sensitivity, with the target set to infinity to encourage maximal improvement. Detector quality is expressed relative to existing detector sensitivities, yielding an interpretable scalar value. Further feasibility terms penalize laser powers above specified thresholds, implemented using rectified linear unit functions. To enable unconstrained optimization within parameter ranges, we reparameterize each variable: unconstrained parameters are mapped to [0, 1] with a sigmoid and then scaled to their individual ranges before simulation.

AI exploration

The expressive interferometer grid ansatz enables continuous exploration using Adam[105] with Differometor's gradients, although a dedicated algorithmic competition could yield better search methods. This produces highly sensitive detectors but the initial ansatz requires simplification. Integrated simplification is difficult because optical losses in mirror components cannot generally be phased out continuously. We therefore use iterative simplification, removing components until detector quality falls below a threshold. Restarting the exploration from simplified detectors can further improve sensitivity.

expressive search space but lowers exploration complexity and ensures that discovered experiments are easily implementable[4,57–60]. Examples are quantum optics experiments in which the splitting ratio of a beam splitter is conventionally fixed at 50%, as tuning this parameter usually requires the implementation of an interferometer[4].

The alternative strategy is to treat parameters as variables to be optimized. This increases exploration complexity but expands the search space with interesting solutions that may outperform state-of-the-art experiments[6,23,25]. Parameter ranges must often be restricted to ensure that corresponding components can actually be manufactured[23,38,51]. An example would be a maximum laser power that should not exceed a certain threshold.

Examples of optimizable parameters are material properties such as permittivity or optical nonlinearities[48], component parameters such as reflectivity, suspension properties or electrical resistance[23,51] and spatial parameters such as distances between components[21].

Physics simulators

Physics simulators enable AI-driven exploration by allowing large numbers of experiments to be evaluated computationally, without requiring time- and cost-intensive realization. They map experiments to their outcomes by solving fundamental physics equations, such as the Schrödinger or Maxwell equations, and can rely on stochastic processes. Unlike prominent deep-learning schemes, which rely on large datasets, physics simulators enable exploration that starts without initial data. The algorithm can use the simulator on demand to actively decide which experiments to evaluate.

The quality of the simulator determines the reliability of discovered designs. As algorithms optimize objectives based on simulator outputs, they will systematically exploit any inaccuracies or bugs of the simulator, producing designs that seem exceptional in simulation but perform poorly after realization. For this reason, proposed designs must be critically examined beyond simulator outputs before committing resources to their physical implementation.

When designing an experiment, the simulator is called many times and typically becomes the computational bottleneck. Beyond domain-specific fast simulation techniques, advances in AI offer general strategies to speed up simulators in optimizations. We discuss details of prominent techniques, including machine learning surrogates, automatic differentiation and other hardware- and software-based acceleration (Fig. 3).

Machine learning surrogates

One strategy is to replace or supplement simulators with machine learning models that are trained to approximate the simulator. Such

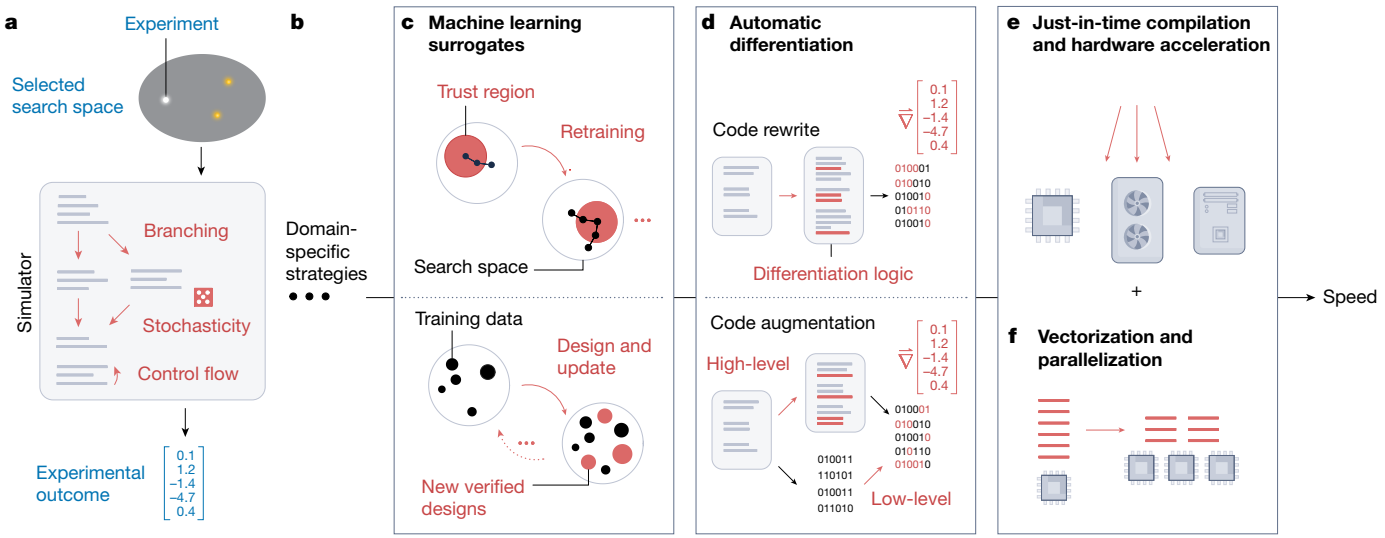


Fig. 3 | Acceleration strategies for physics simulators in optimization workflows. Red highlights important features and blue indicates the inputs and outputs of the simulator. **a**, Simulators map experiments from an engineered search space to experimental outcomes and may include stochastic components, complex branching logic and control flow statements. **b**, Besides many domain-specific strategies that depend on the simulator and scientific context, there are four general acceleration strategies. **c**, Machine learning surrogates provide fast approximations to simulator outputs. Because exhaustive training over the full search space is infeasible, models are trained either locally within trust regions and retrained as exploration progresses or iteratively on an expanding set of surrogate-proposed, simulator-verified designs. **d**, Automatic differentiation enables efficient gradient computation for gradient-based optimization, either by rewriting simulators in differentiable frameworks or by using code augmentation tools that insert differentiation logic into high-level source code or low-level code representations. **e**, Just-in-time compilation and hardware acceleration improve runtime performance by adapting compiled code to execution patterns and using specialized hardware. **f**, Vectorization and parallelization restructure computations to execute simultaneously across data elements or hardware resources.

surrogates can reach orders-of-magnitude speed-up in each iteration—even considering training and data generation time[61–63]. They can guide gradient-based designs by supporting gradient computation[43,64–67], serve as correction functions to simulator approximations[68] or reduce the number of expensive simulations when objective functions are based on statistically rare events[69]. This approach is useful when approximate physics is acceptable and speed matters more than accuracy. When the simulator is complex, poorly documented or closed source, machine learning surrogates allow it to be treated as a black box and replaced with a faster approximation.

The development of surrogate models requires special treatment of intrinsic uncertainties[70]. Especially, the use of surrogates outside their training distribution leads to substantial loss of accuracy[43]. Training surrogates to cover the entire search space would greatly increase data requirements and is infeasible, as regions of interest typically contain only a few known designs. There are common strategies to circumvent this (Fig. 3). First, by approximating the simulator only within a local neighbourhood of samples and retraining surrogates on a new neighbourhood once the exploration moves beyond this trust region[43,65,66]. Second, by training surrogates on random designs and retraining them as newly discovered, simulator-verified designs are iteratively added to the dataset[64]. Both approaches allow exploration with partially trained surrogates while reducing the risk of unreliable predictions.

A creative, albeit organizationally demanding, way to address the high computational cost of training surrogates and exploring design choices is through community challenges. For example, the Calorimeter Simulation Challenge attracted 59 distinct submissions[71], yielding a surrogate diversity that would be infeasible for an individual researcher to explore.

Automatic differentiation

Many exploration methods rely on gradients that quantify how experimental performance changes as parameters are varied. Yet many simulators include stochastic components, branching logic and are implemented in programming languages that provide no analytic derivatives. If numerical differentiation is possible at all, it is an inefficient default, as each gradient requires one simulator evaluation per parameter. Surrogates can be used as differentiable simulators but they remain approximations. For fast gradient computation with exact physics, simulators can be equipped with chain-rule-based automatic differentiation, either by rewriting the simulator code in an automatic differentiation framework or by using tools that automatically augment the simulator with gradient logic (Fig. 3).

Popular automatic differentiation frameworks such as JAX[72], PyTorch[73] or TensorFlow[74] compute gradients by tracing program execution and applying the chain rule to the resulting computational graph. Recent work across several areas of physics has demonstrated that simulators implemented using such frameworks can achieve orders-of-magnitude speed-ups in gradient computation[25,45,47,75–77]. However, the tracing mechanism imposes constraints on common programming patterns—for example, if statements or stochastic processes are not available in their general form. Rewriting a simulator using an automatic differentiation framework can therefore require notable development efforts to satisfy these constraints while preserving the underlying physics. This overhead represents a substantial barrier to adoption for large simulators, although it may become less important as large language models (LLMs) continue to advance in programming tasks[78,79].

An alternative to full rewrites are code augmentation tools, which transform simulator code into a new program augmented with gradient logic. Augmentation tools for languages such as Julia or C (refs. 80,81) have recently been used for the implementation of differentiable physics simulators[82–84]. Other tools can augment even on the machine code level[85] but typically offer low performance[84]. They are often used for exploratory differentiation, such as first attempts to differentiate extensive physics simulators such as GEANT4 (refs. 84,86).

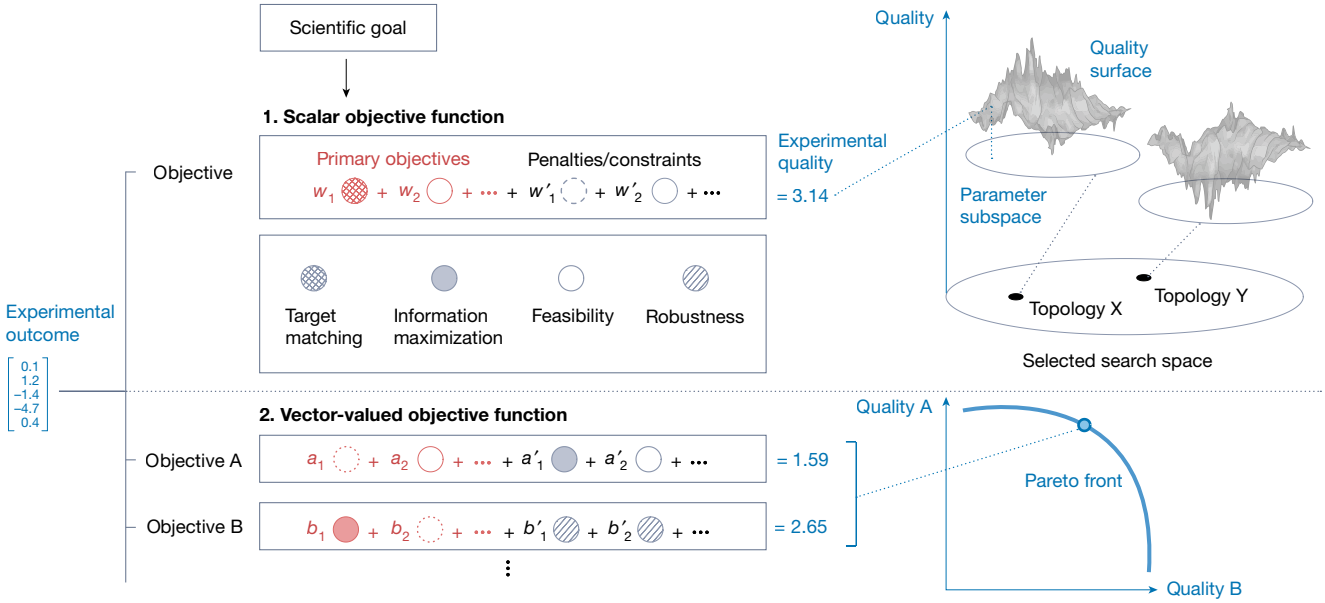


Fig. 4 | Scalar and vector-valued objective functions. Red highlights primary objectives and blue indicates inputs and outputs of the objective function. (1) A scalar objective maps the experimental outcome from the simulator to a single quality value, defining a quality surface over the selected search space. This surface may contain distinct subsurfaces corresponding to individual topologies or discrete parameter sets. Objectives are often composed of weighted primary terms together with penalties and constraints. (2) Vector-valued objectives evaluate several quality metrics simultaneously. The resulting trade-off surface forms a Pareto front, a set of solutions in which improving one objective worsens at least one other.

Automatic differentiation can yield misleading gradients[84], which commonly occurs in simulators with discrete randomness, such as Monte Carlo sampling or conditional logic[47,87,88]. In such cases, gradients may point in directions that do not correspond to real performance improvements, making careful validation of gradient behaviour essential before relying on gradient-based optimization[89].

Other hardware and software acceleration

Modern differentiation frameworks offer hardware acceleration[90], machine learning compilers[91] and just-in-time compilation[92], which tailors compilation to runtime data rather than relying on static assumptions. They also support automatic vectorization, the automated transformation of scalar functions into vectorized operations or parallelization of workloads on several hardware devices. With data structures such as large grids, these features can contribute to orders-of-magnitude speed-ups[25].

When simulators are replaced by surrogates or rewritten within an automatic differentiation framework, many of these acceleration features become available with little extra effort, making such approaches an attractive way to benefit from rapid advances in AI.

Objective functions

In AI-driven design of physics experiments, the difficulty of asking the right question manifests itself in two challenges: defining a scientific goal and translating it into an efficiently computable objective function. The definition of scientific goals is primarily driven by human interest, resource allocations or practical needs. The second challenge of translating goals into functions is the focus of this section. Typically, an objective function takes the simulated outcome of a proposed experiment as input and computes the quality of this experiment as a scalar value, thereby defining a quality surface over the search space (Fig. 4). This surface can accommodate distinct subsurfaces, each corresponding to a different topology or discrete parameter set. Often, it is highly irregular and the number of local optima grows rapidly with the component number[93]. Similar to how neural network architectures influence their loss surfaces[94], the approximations used in the physics simulator and the encoding of components, topology and parameters shape the quality surface in experiment design tasks.

The efficiency with which an objective function can be evaluated depends on its formulation, the experiment definition and the simulator's acceleration strategies. An objective function can be scalar or vector-valued.

Scalar objective functions

A scalar objective can be formulated as a weighted sum of several, possibly conflicting, terms[6,23,95,96], for example, sensitivity and robustness (Fig. 4). Primary objectives define the main goals of the design, sometimes highest sensitivity, but sometimes also low complexity or robustness. Penalty and constraint terms encode the physical, engineering or practical conditions under which these goals must be achieved, for example, staying above a certain sensitivity threshold.

Target matching terms apply when an experiment aims to reproduce a concrete target, such as a specific quantum state[6,97] or known particle energies reconstructed by a detector[65,66]. Experiment quality is then measured by the distance to this target, for example, using the average over squared differences, a common metric in machine learning regression problems. Estimating this distance may require nontrivial readout of simulator outputs, such as inferring particle energies from energy deposits across detector layers. Because readout quality can depend on the detector topology, we may need either a generic readout scheme or joint optimization of the readout and detector hardware[98].

Information maximization terms remove the need for specifying a concrete target or readout and instead enable the design of experiments whose outcomes are maximally informative about unknown quantities of interest. Common choices include Fisher information and mutual information, which facilitate comparisons to information-theoretic limits. Such terms have been used to design quantum states with enhanced sensitivity[96,99], high-energy detectors that retain maximal information about particle energies in their responses[65] and optimal array configurations for neutrino detectors[21] and radio interferometers[41].

Feasibility terms ensure that experiment designs can be realized under given constraints. For example, monetary penalties for magnetic components enforce economic feasibility[17]. Penalties on coil lengths

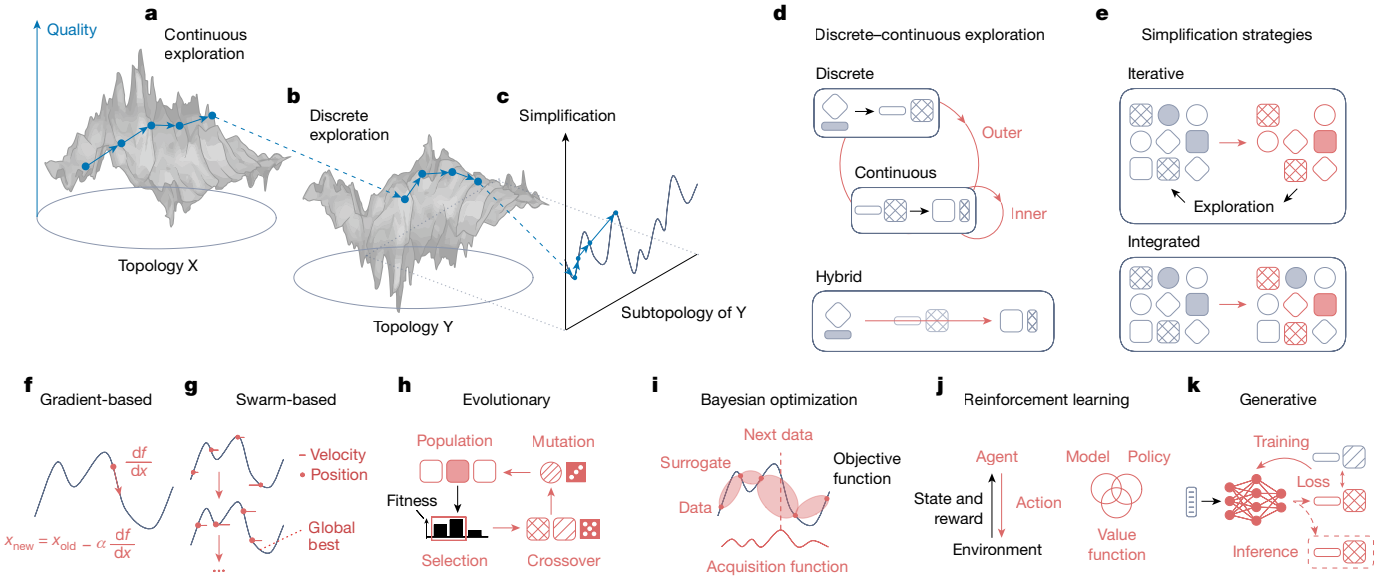


Fig. 5 | Exploration challenges and algorithms for physics experiment design. Red highlights important algorithmic features and blue indicates inputs and outputs of the exploration. **a**, Continuous exploration within the continuous parameter space of a fixed experiment topology. **b**, Discrete exploration across distinct topologies or discrete parameter sets. **c**, AI-based designs may be overparameterized and require simplification to lower-dimensional representations. **d**, Continuous and discrete exploration can be performed together through nested loops or hybrid methods that directly integrate both. **e**, Simplification can be integrated directly into optimization by continuous deactivation of components or performed iteratively by alternating component deletion and re-optimization. **f**–**k**, Representative exploration methods. **f**, Gradient-based methods choose higher-quality experiments based on the gradient of the objective function with respect to the parameters. **g**, Swarm-based algorithms guide many simple agents that share information to collectively explore a search space and move towards high-quality experiments based on shared performance cues. **h**, Evolutionary algorithms evolve a population of candidate experiments by selection, random crossover and random mutation, improving them over generations based on a fitness function. **i**, Bayesian optimization selects new experiments by modelling the objective function with a probabilistic surrogate and using an acquisition function that chooses the designs with the highest expected gain under uncertainty. **j**, Reinforcement learning interacts with an environment, using a policy that maps states to actions, a value function that estimates long-term returns and optionally a model of the environment, and it improves its actions through reward-driven feedback. **k**, Generative models learn a data distribution and produce new designs by modelling or transforming latent variables so that generated outputs resemble the training data.

in fusion reactors ensure manufacturability[15,100] and component count minimization terms reduce resource use in quantum circuits[101,102]. Penalties on the laser power prevent damage in gravitational wave detectors[23] and minimum feature size requirements enforce fabrication constraints in nanophotonic devices[49].

Robustness terms guarantee that experiments are insensitive to unwanted background effects, manufacturing imperfections or changes in operating conditions. Event-count minimization improves neutron moderator designs that suppress statistically rare background events[69]. Gradient- or perturbation-based penalties make mirror coatings[103] or squeezing performance in gravitational wave detectors[95] robust against manufacturing errors and Monte Carlo evaluations minimize performance variance of superconducting circuit designs under fabrication perturbations[38].

Understanding the effects of these terms on the quality surface through visualization and analysis[93,94,104] is a key challenge to prevent excessive local optima or ruggedness.

Vector-valued objective functions

Vector-valued objectives originate from multipurpose experiments or engineering trade-offs that define a Pareto front—a set of solutions in which improving one objective inevitably makes other objectives worse[44,100]. Balancing particle momentum and direction measurement precision with trajectory reconstruction efficiency results in a Pareto front of high-energy detectors[44]. Similarly, stellarator coil optimizations use bi-objective formulations to trade off coil simplicity against magnetic field error[100].

Although these approaches are computationally demanding, they provide a diverse set of solutions from which appropriate configurations can be selected later, rather than forcing convergence to a single compromise. Fully exploring Pareto fronts, instead of collapsing the objectives into a weighted scalar formulation, is therefore essential to fully exploit this methodology.

AI exploration

AI-based exploration methods aim to find high-quality experiments within a search space. They take experiments and their corresponding quality values as input and propose new designs to improve the overall quality. Three main challenges arise: continuous exploration, discrete exploration and simplification. Each of these challenges can be addressed by different classes of algorithms, summarized in Fig. 5.

Continuous exploration

Experiment design simplifies to pure continuous exploration whenever topologies are fixed[20,43,67,95], can be tuned continuously[21,47] or are embedded within an expressive, continuous ansatz from which smaller experimental topologies can be extracted[6].

Continuous exploration can use either gradient-based or gradient-free methods. Gradient-based optimizers such as Adam[105] use valuable gradient information to follow the direction of steepest improvement and scale to billions of parameters. Although the use of stochastic gradients can help escape shallow local optima, such methods primarily explore the local neighbourhood of their starting position. Gradients can be computed in various ways, including analytical or automatic differentiation, machine learning surrogates, numerical differentiation or stochastic gradient estimators such as reparameterization methods for continuous random variables[84] and score-function estimators for

discrete variables[47]. Gradient-based methods discovered new quantum optics experiments[6,106], nanophotonic devices[49,50], magnetic surfaces for stellarators[16,107], super-resolution microscope blueprints[25] and interesting parameter configurations for high-energy physics experiments[43,65,66,108,109].

Continuous gradient-free exploration methods such as Bayesian optimization and evolutionary or swarm-based algorithms (Fig. 5) often trade computational efficiency for a higher likelihood of finding a global optimum. Although their performance typically declines as dimensionality increases, continuing research steadily extends their applicability to problems with thousands of parameters[110]. They are used when gradients are not available or many low-quality local optima exist, for example, in specific cases of stellarator coil design[111], gravitational wave detectors[46,95,103] or high-energy physics experiments[17,18,44,112]. Often, a combination of gradient-based and gradient-free algorithms over several optimization stages is used to balance advantages and disadvantages of both approaches, for example, by initializing starting positions of gradient-based methods with a few steps of Bayesian optimization[111].

Discrete exploration

Discrete exploration is required when experimental design involves switching between distinct topologies or discrete parameter sets. Exploration is fully discrete when the search space is composed entirely of components without continuous parameters or when continuous parameters are fixed.

Generative models (Fig. 5) can perform discrete exploration by learning to generate symbolic descriptions of experimental set-ups, for example, in the form of Python code[59] or vector representations[113], and have uncovered previously unknown set-ups for several quantum state families. Similarly, reinforcement learning agents (Fig. 5) can design discrete quantum protocols or circuits by learning strategies to sequentially place suitable components[60,114]. Many other algorithms that can perform discrete exploration are mostly used in combination with continuous exploration, as discussed in the next section.

Discrete exploration can be improved by learning so-called gadgets, small discrete functional units that are iteratively discovered and reused to assemble increasingly complex experiments[4,57,115–117]. Although algorithmically more demanding, gadget learning can speed up the exploration process by orders of magnitude.

Discrete–continuous exploration

In the most general case, designing an experiment from scratch requires both switching between discrete topologies or parameter sets and tuning continuous parameters. This challenging, discrete–continuous exploration is commonly solved through two different approaches: nested optimization loops, in which continuous and discrete exploration are performed iteratively, or hybrid methods that integrate both exploration types directly (Fig. 5).

Outer discrete exploration loops often use reinforcement learning[118,119] or evolutionary strategies[39] to find interesting topologies that are then optimized with local, continuous exploration in an inner loop. Even with an overcomplete ansatz, the presence of many local optima makes it beneficial to complement gradient-based methods with an outer discrete exploration loop that iterates over different ansatz configurations[23]. Augmenting the discrete loops with domain knowledge[6,8,42,120] or known reference designs[121] can simplify the search and lead to pronounced speed-up.

Some hybrid methods use an expressive ansatz that transforms a discrete–continuous problem into a purely continuous one by embedding discrete topologies within its continuous parameters[6,23,25,106]. Evolutionary algorithms can encode discrete and continuous parameters within a single gene and have been used for the hybrid design of quantum metrology experiments[5,97,99]. In high-energy physics, reinforcement learning agents can handle hybrid action spaces of continuous positions and discrete sensor choices[52], whereas gradient-based methods can perform hybrid exploration when relaxing categorical choices into continuous vectors that can be evaluated with a trained surrogate model[66]. Finally, diffusion models can have coupled diffusion processes for both discrete component choices and continuous parameters simultaneously, enabling quantum circuit design through generative models[7,122].

Developing new exploration strategies and representations capable of jointly managing discrete–continuous search spaces remains a key challenge. Given the recent success of LLMs, an open question is whether such models can augment hybrid methods with reasoning abilities, for instance, as high-level design planners[123], by guiding evolutionary algorithms[124], through agentic systems[125] or through large physics models[126].

Simplification

AI-based design frequently yields overparameterized set-ups in which some components or parameters contribute negligibly to performance. Such redundancies enable substantial simplification: components can be removed or parameters fixed to simpler values without substantially degrading experimental quality, effectively descending from a higher- to a lower-dimensional quality surface (Fig. 5).

Sometimes, simplification can be integrated directly into the exploration process by allowing parameters to continuously deactivate components—for example, by driving the power of a laser to zero. Such integrated simplification has been observed in the design of super-resolution microscopes[25] and can be further encouraged through regularization terms in the objective function that bias the exploration towards simple or inactive values. In other cases, simplification is performed iteratively alongside exploration: unnecessary components are removed according to logic-based rules[53] or importance thresholds[6,23,106], the resulting designs are re-optimized and the cycle is repeated until further simplification leads to an unavoidable loss in quality.

In general, effective simplification often relies on substantial domain knowledge and a detailed understanding of the proposed designs. How such interpretive steps can be automated remains an important open question for translating discovered solutions into realizable experiments.

The vision of de novo discovery

AI can turn experiment design into an explicit search problem and offers the opportunity to find new measurement techniques that reveal new insights into the fundamental rules of the Universe. By broadening the space of candidate designs, AI can accelerate the exploration phase of experimental design cycles, in which early choices often determine scientific reach for decades. So far, only a few AI-designed experiments have been implemented[9–11]. Notable barriers include long realization cycles, difficulty interpreting complex designs and the need to develop new methods to implement unconventional yet high-quality experiments. Other AI-proposed designs have led to new experimental insights[12–14] or may influence next-generation experiments[16,18,23].

To further explore the potential of this new design pathway, substantial challenges remain to be solved: the engineering of new, clever experiment search spaces that simplify exploration without losing expressibility; the implementation of simulators or physics approximations that support the exploration through fast design evaluation; the formulation of efficiently computable objective functions that are aligned with scientific goals and result in smooth quality surfaces; the invention of new exploration algorithms that can navigate high-dimensional, discrete–continuous search spaces; and, finally, the development of discovery frameworks that combine all of these steps into easy-to-use libraries[6,108,127].

Advancing on these challenges might bring us closer to AI-driven design in its most extreme form—de novo discovery.

Foundational simulator

Imagine a foundational simulator: a single program capable of modelling experiments across several physical domains and scales. Simulators for individual fields of physics already allow for new experimental designs. However, integrating these simulators into a single foundational simulator would allow algorithms to uncover design principles or effects from vastly different fields of physics that are rarely considered together.

For example, a simulator for electron beam physics can be used for the discovery of new electron microscope technology. Integrating these physics into a general simulator that also includes the physics of gravitational wave interferometry would allow AI exploration that exploits complex electron-optical dynamics—which is not possible with two isolated simulators. Likewise, such a foundational simulator might enable gravitational wave researchers to discover detector concepts that draw on electron beam physics in unexpected ways.

Realizing this vision requires new representations and approximations that remain accurate and computationally tractable across many physical scales. Early examples can be seen in fusion physics, in which simulations span several orders of magnitude in space and time[128], and photonics, in which simulators cover wide ranges of coupled optical, electrical, thermal and material effects[129].

Large objective space

Imagine further that we go beyond individual objective functions and develop lists of hundreds of interesting physical objectives, for example, systematic generalizations of our best physics theories. Consider the relation between kinetic energy and velocity, $E = \frac{mv^2}{2}$. Now imagine a nineteenth-century scientist wondering why the dependence is quadratic and suggesting a different form,

$$E = \frac{m}{2}v^2 + \alpha v^3 + \beta v^4 + O(v^5). \quad (1)$$

Physical theories of the nineteenth century would predict $\alpha = 0 = \beta$. Yet, equation (1) yields an interesting objective: designing experiments that are particularly sensitive to shifts of α and β from zero. Such experiments would have revealed that α is indistinguishable from zero but β is not (owing to special relativity, $\beta = \frac{3m}{8c^2}$). This would have been a critical indication for exploring new science—an opportunity to glimpse effects of special relativity well before 1905.

Parametric extensions of modern physics theories already exist. A prominent example is Kostelecký's Standard-Model Extension, which introduces 155 parameters in its minimal form[130,131]. Each parameter in this framework—and similarly in others such as post-Newtonian parameterizations[132]—defines a distinct objective for a wide range of experiments. Coupling this with a foundational simulator and AI-based exploration could result in an AI-generated experimental test bed for theories in fundamental physics.

Experimental groups could constrain the search space to their individual capabilities and discover experiments that can be performed with experimental components in their laboratory. Many candidate experiments might operate at much smaller scales than those pursued by large collaborations—for example, tabletop experiments[133] including atom interferometry[134], quantum optomechanics[135], quartz oscillators[136], large quantum systems[137] and spectroscopy of atoms and molecules[138]—leading to a democratization of fundamental physics discoveries.

Some aspects of the objective formulation may also increasingly be supported by AI. Early prototypes based on LLMs and knowledge graphs show promise of complementing human ideas by autonomously generating scientific goals[139,140] or by forecasting impactful research directions[141,142].

Experiment discovery as an AI benchmark

If the search space, simulator and objective functions are provided as black-box interfaces, the AI community does not require extensive physics knowledge but can treat experiment design as an exceptionally demanding and impactful benchmark[143,144], advancing algorithmic research while addressing critical physics questions. Establishing a fair testing ground, for instance, through AI competitions[71], would empower the AI community to contribute to fundamental physics questions, potentially leading to large interdisciplinary collaborations between physics and AI research.

The impact of modern AI and LLMs

Although many design tasks can already be tackled with traditional optimization techniques such as genetic algorithms or Bayesian optimization, modern AI and deep learning are becoming essential as search spaces grow more expressive and simulators more complex. These frameworks enable fast, data-driven approximations and learning-based generative models that go beyond direct exploration. For example, neural surrogate models can approximate slow simulators[64,71] and neural quantum states can approximate large physical systems[145]. In the computational exploration of design spaces, deep reinforcement learning and generative models (including variational autoencoders and diffusion models) are increasingly being applied[52,113,122], and the community is still exploring which techniques and combinations thereof work best for practical problems.

Although LLMs have an enormous impact on the lives of many of us, their impact on physics experiment design has so far been limited to a few proof-of-concept demonstrations[123,125]. We expect this to change substantially as frontier models continue to improve in math, coding and physics knowledge. Looking ahead, LLMs could help expand or better represent human-defined search spaces, accelerate the development of modern auto-differentiating simulators and support or fully automate the objective formulation by proposing new physical targets[139]. They may also be incorporated with other exploration algorithms to suggest promising starting points and interesting regions of the search space to explore (or to avoid). Overall, we expect progress to come from a deep integration of classical optimization, modern AI methods and frontier LLMs, rather than from any single technique alone.

Looking ahead

Realizing the vision of de novo experiment discovery will require advances on many fronts. In physics, new efficient representations and reliable approximations will be necessary to build foundational simulators spanning several fields of physics. Objective spaces for potential benchmarks need to be carefully selected and formulated in standardized formats, easy to use without extensive background knowledge. In computer science, more powerful exploration algorithms will be required to navigate increasingly large discrete–continuous search spaces. Finally, whenever an algorithm discovers an experiment that outperforms human designs, the solution necessarily contains new ideas, experimental tricks or concepts that humans have not been aware of. To avoid a future in which physics experiments can be performed but not comprehended, there will be an increasing need for powerful tools with which future physicists can extract and understand artificial insights[146,147]. Beyond the usefulness of superior designs for observing interesting physics, the discovered designs can then inspire the development of new experimental techniques and concepts.

Ultimately, this vision augments the role of the human scientist—from an intuition-guided designer to an orchestrator of intelligent automated exploration.

1. Franklin, A. *Experiment, Right or Wrong* (Cambridge Univ. Press, 1990).
2. Radder, H. *The Philosophy of Scientific Experimentation* (Univ. Pittsburgh Press, 2003).
3. Shamos, M. H. *Great Experiments in Physics: Firsthand Accounts from Galileo to Einstein* (Courier Corporation, 1987).

4. Krenn, M., Malik, M., Fickler, R., Lapkiewicz, R. & Zeilinger, A. Automated search for new quantum experiments. *Phys. Rev. Lett.* **116**, 090405 (2016).
This paper presents Melvin, an early AI-driven framework for the design of photonic quantum information experiments, with several of its proposed experimental configurations subsequently realized in laboratories.
5. Knott, P. A search algorithm for quantum state engineering and metrology. *New J. Phys.* **18**, 073033 (2016).
This paper presents Tachikoma, an early AI-driven framework for designing quantum metrology experiments, initially using genetic algorithms and later incorporating neural network surrogate models.
6. Ruiz-Gonzalez, C. et al. Digital discovery of 100 diverse quantum experiments with PyTheus. *Quantum* **7**, 1204 (2023).
This paper presents PyTheus, a discovery framework for photonic quantum optics that enables efficient exploration of experimental designs via an overcomplete, physics-inspired continuous search space, yielding 100 diverse quantum experiments.
7. Fürrutter, F., Muñoz-Gil, G. & Briegel, H. J. Quantum circuit synthesis with diffusion models. *Nat. Mach. Intell.* **6**, 515–524 (2024).
8. Landgraf, J., Wanjura, C. C., Peano, V. & Marquardt, F. Artificial discovery of lattice models for wave transport. Preprint at https://doi.org/10.48550/arXiv.2508.10693 (2025).
9. Babazadeh, A. et al. High-dimensional single-photon quantum gates: concepts and experiments. *Phys. Rev. Lett.* **119**, 180510 (2017).
10. Erhard, M., Malik, M., Krenn, M. & Zeilinger, A. Experimental Greenberger–Horne–Zeilinger entanglement beyond qubits. *Nat. Photonics* **12**, 759–764 (2018).
11. Qian, K. et al. Multiphoton non-local quantum interference controlled by an undetected photon. *Nat. Commun.* **14**, 1480 (2023).
12. Krenn, M., Hochrainer, A., Lahiri, M. & Zeilinger, A. Entanglement by path identity. *Phys. Rev. Lett.* **118**, 080401 (2017).
13. Wang, K. et al. Entangling independent particles by path identity. *Phys. Rev. Lett.* **133**, 233601 (2024).
14. Arlt, S., Ruiz-Gonzalez, C. & Krenn, M. Emulating multiparticle emitters with pair-sources: digital discovery of a quantum optics building block. *Quantum Sci. Technol.* **10**, 015042 (2024).
15. Paul, E., Landreman, M., Bader, A. & Dorland, W. An adjoint method for gradient-based optimization of stellarator coil shapes. *Nucl. Fusion* **58**, 076015 (2018).
16. Goodman, A. G. et al. Quasi-isodynamic stellarators with low turbulence as fusion reactor candidates. *PRX Energy* **3**, 023010 (2024).
17. Baranov, A. et al. Optimising the active muon shield for the SHiP experiment at CERN. *J. Phys. Conf. Ser.* **934**, 012050 (2017).
18. Cisbani, E. et al. AI-optimized detector design for the future Electron-Ion Collider: the dual-radiator RICH case. *J. Instrum.* **15**, P05009 (2020).
This paper demonstrates the use of Bayesian optimization to improve the design of a large-scale particle physics detector within a fixed topology search space.
19. Dorigo, T. et al. Toward the end-to-end optimization of particle physics instruments with differentiable programming. *Rev. Phys.* **10**, 100085 (2023).
20. Rolla, J. et al. Using evolutionary algorithms to design antennas with greater sensitivity to ultrahigh energy neutrinos. *Phys. Rev. D* **108**, 102002 (2023).
This paper demonstrates AI-driven design of highly sensitive ultrahigh energy neutrino antennas using evolutionary algorithms within a fixed topology search space.
21. Haack, C. et al. Machine-learning aided detector optimization of the Pacific Ocean Neutrino Experiment. In *Proc. 38th International Cosmic Ray Conference (ICRC2023)* id.1059 (2024).
22. Dorigo, T. et al. Toward the end-to-end optimization of the SWGO array layout. *Nucl. Phys. B* **1017**, 116934 (2025).
23. Krenn, M., Drori, Y. & Adhikari, R. X. Digital discovery of interferometric gravitational wave detectors. *Phys. Rev. X* **15**, 021012 (2025).
This paper demonstrates that AI-driven exploration of overcomplete interferometric topology search spaces can discover novel gravitational wave detector topologies that outperform current next-generation designs under realistic conditions.
24. Nehme, E. et al. DeepSTORM3D: dense 3D localization microscopy and PSF design by deep learning. *Nat. Methods* **17**, 734–740 (2020).
25. Rodríguez, C., Arlt, S., Möckl, L. & Krenn, M. Automated discovery of experimental designs in super-resolution microscopy with XLuminA. *Nat. Commun.* **15**, 10658 (2024).
26. Huang, P.-S., Boyken, S. E. & Baker, D. The coming of age of de novo protein design. *Nature* **537**, 320–327 (2016).
27. Sanchez-Lengeling, B. & Aspuru-Guzik, A. Inverse molecular design using machine learning: generative models for matter engineering. *Science* **361**, 360–365 (2018).
28. Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O. & Walsh, A. Machine learning for molecular and materials science. *Nature* **559**, 547–555 (2018).
29. Sigmund, O. & Maute, K. Topology optimization approaches: a comparative review. *Struct. Multidiscip. Optim.* **48**, 1031–1055 (2013).
30. Duris, J. et al. Bayesian optimization of a free-electron laser. *Phys. Rev. Lett.* **124**, 124801 (2020).
31. Vendeiro, Z. et al. Machine-learning-accelerated Bose-Einstein condensation. *Phys. Rev. Res.* **4**, 043216 (2022).
32. Degrave, J. et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature* **602**, 414–419 (2022).
33. Reinschmidt, M., Fortágh, J., Günther, A. & Volchkov, V. V. Reinforcement learning in cold atom experiments. *Nat. Commun.* **15**, 8532 (2024).
34. Buchli, J. et al. Improving cosmological reach of a gravitational wave observatory using Deep Loop Shaping. *Science* **389**, 1012–1015 (2025).
35. Rainforth, T., Foster, A., Ivanova, D. R. & Bickford Smith, F. Modern Bayesian experimental design. *Stat. Sci.* **39**, 100–114 (2024).
36. DeZoort, G., Battaglia, P. W., Biscarat, C. & Vlimant, J.-R. Graph neural networks at the Large Hadron Collider. *Nat. Rev. Phys.* **5**, 281–303 (2023).
37. Dax, M. et al. Real-time inference for binary neutron star mergers using machine learning. *Nature* **639**, 49–53 (2025).
38. Rajabzadeh, T., Boulton-McKeehan, A., Bonkowsky, S., Schuster, D. I. & Safavi-Naeini, A. H. A general framework for gradient-based optimization of superconducting quantum circuits using qubit discovery as a case study. Preprint at https://doi.org/10.48550/arXiv.2408.12704 (2024).
39. MacLellan, B. et al. Inverse design of photonic systems. *Laser Photonics Rev.* **18**, 2300500 (2024).
40. Crafts, J., Fatemi, R., Horn, T. & Kalinkin, D. Performance optimization for a scintillating glass electromagnetic calorimeter at the EIC. *J. Instrum.* **19**, C05049 (2024).
41. Wang, J., Bacharach, L., Larzabal, P. & El Korso, M. N. A comparison of antenna placement criteria based on the Cramér–Rao and Barankin bounds for radio interferometer arrays. *Signal Process.* **219**, 109404 (2024).
42. Miao, H., Yang, H., Adhikari, R. X. & Chen, Y. Quantum limits of interferometer topologies for gravitational radiation detection. *Class. Quantum Gravity* **31**, 165010 (2014).
43. Shirobokov, S., Belavin, V., Kagan, M., Ustyuzhanin, A. & Baydin, A. G. Black-box optimization with local generative surrogates. *Adv. Neural Inf. Process. Syst.* **33**, 14650–14662 (2020).
This paper introduces a trust-region-constrained generative surrogate model as a novel optimization method for design tasks with black-box stochastic simulators.
44. Fanelli, C. et al. AI-assisted optimization of the ECCE tracking system at the Electron Ion Collider. *Nucl. Instrum. Methods Phys. Res. A* **1047**, 167748 (2023).
45. Schillings, P. & Erdmann, J. Fighting Newtonian noise with gradient-based optimization at the Einstein Telescope. *Class. Quantum Gravity* **42**, 065025 (2025).
46. Badaracco, F., Harms, J. & Rei, L. Joint optimization of seismometer arrays for the cancellation of Newtonian noise from seismic body waves in the Einstein telescope. *Class. Quantum Gravity* **41**, 025013 (2024).
47. Kagan, M. & Heinrich, L. Branches of a tree: taking derivatives of programs with discrete and branching randomness in high energy physics. Preprint at https://doi.org/10.48550/arXiv.2308.16680 (2023).
48. Hughes, T. W., Minkov, M., Williamson, I. A. & Fan, S. Adjoint method and inverse design for nonlinear nanophotonic devices. *ACS Photonics* **5**, 4781–4787 (2018).
49. Sapra, N. V. On-chip integrated laser-driven particle accelerator. *Science* **367**, 79–83 (2020).
This paper demonstrates the AI-driven discovery of an on-chip integrated laser-driven particle accelerator by exploring a parametric topology search space encoded as a permittivity distribution.
50. Huang, S.-Y. et al. Multiphoton quantum interference at ultracompact inverse-designed multiport beam splitter. *Opt. Quantum* **3**, 576–582 (2025).
51. Menke, T. et al. Automated design of superconducting circuits and its application to 4-local couplers. *npj Quantum Inf.* **7**, 49 (2021).
This paper demonstrates AI-driven discovery of novel superconducting quantum circuits through automated search over a slot-based topology search space.
52. Qasim, S. R., Owen, P. & Serra, N. Physics instrument design with reinforcement learning. *Mach. Learn. Sci. Technol.* **6**, 035033 (2025).
53. Cervera-Lierta, A., Krenn, M. & Aspuru-Guzik, A. Design of quantum optical experiments with logic artificial intelligence. *Quantum* **6**, 836 (2022).
54. Liu, H., Simonyan, K. & Yang, Y. Darts: differentiable architecture search. In *Proc. 7th International Conference on Learning Representations (ICLR 2019)* (2019).
55. Zhang, S.-X., Hsieh, C.-Y., Zhang, S. & Yao, H. Differentiable quantum architecture search. *Quantum Sci. Technol.* **7**, 045023 (2022).
56. Sim, S., Johnson, P. D. & Aspuru-Guzik, A. Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms. *Adv. Quantum Technol.* **2**, 1900070 (2019).
This paper proposes practical metrics for evaluating the expressibility of parameterized quantum circuits, informing the design of overcomplete topology search spaces.
57. Olle, J., Yevtushenko, O. M. & Marquardt, F. Scaling the automated discovery of quantum circuits via reinforcement learning with gadgets. Preprint at https://doi.org/10.48550/arXiv.2503.11638 (2025).
58. Mirhoseini, A. et al. A graph placement methodology for fast chip design. *Nature* **594**, 207–212 (2021).
59. Arlt, S. et al. Meta-designing quantum experiments with language models. *Nat. Mach. Intell.* **8**, 148–157 (2026).
60. Wallnöfer, J., Melnikov, A. A., Dür, W. & Briegel, H. J. Machine learning for long-distance quantum communication. *PRX Quantum* **1**, 010301 (2020).
61. Edelen, A. et al. Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems. *Phys. Rev. Accel. Beams* **23**, 044601 (2020).
62. Kasim, M. F. et al. Building high accuracy emulators for scientific simulations with deep neural architecture search. *Mach. Learn. Sci. Technol.* **3**, 015013 (2021).
63. Citrin, J. et al. Fast transport simulations with higher-fidelity surrogate models for ITER. *Phys. Plasmas* **30**, 062501 (2023).
64. Ruiz-Gonzalez, C. et al. Neural surrogates for designing gravitational wave detectors. Preprint at https://doi.org/10.48550/arXiv.2511.19364 (2025).
65. Woźniak, K. A. et al. End-to-end optimal detector design with mutual information surrogates. *Mach. Learn. Sci. Technol.* **6**, 045047 (2025).
This paper demonstrates end-to-end particle physics detector design using a trust-region-constrained surrogate model and a general information-theoretic objective function.
66. Schmidt, K. et al. End-to-end detector optimization with diffusion models: a case study in sampling calorimeters. *Particles* **8**, 47 (2025).
67. Martínez, M. P., Vidal, X. C. & Vischia, P. Automatic optimization of a parallel-plate avalanche counter with optical readout. *Particles* **8**, 26 (2025).
68. Bein, S., Connor, P., Pedro, K., Schleper, P. & Wolf, M. Refining fast simulation using machine learning. In *Proc. 26th International Conference on Computing in High Energy and Nuclear Physics (CHEP 2023)* 09032 (EDP Sciences, 2024).
69. Schuetz, A.-K., Poon, A. W. & Li, A. RESuM: a rare event surrogate model for physics detector design. In *Proc. Thirteenth International Conference on Learning Representations (ICLR 2025)* (2025).

70. Adelmann, A. et al. New directions for surrogate models and differentiable programming for High Energy Physics detector simulation. In *Proc. 2021 U.S. Community Study on the Future of Particle Physics* (2022).
71. Krause, C. et al. CaloChallenge 2022: a community challenge for fast calorimeter simulation. *Rep. Prog. Phys.* **88**, 116201 (2025).
This paper documents the CaloChallenge, a popular competition in the AI and particle physics communities aimed at developing diverse generative models to accelerate calorimeter simulation.
72. Bradbury, J. et al. JAX: composable transformations of Python+NumPy programs. *GitHub* https://github.com/jax-ml/jax (2018).
73. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. *Adv. Neural Inf. Process. Syst.* **32**, 8024–8035 (2019).
74. Abadi, M. et al. TensorFlow: a system for large-scale machine learning. In *Proc. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16)* 265–283 (2016).
75. Kaiser, J., Xu, C., Eichler, A. & Santamaria Garcia, A. Bridging the gap between machine learning and particle accelerator physics with high-speed, differentiable simulations. *Phys. Rev. Accel. Beams* **27**, 054601 (2024).
76. Citrin, J. et al. TORAX: a fast and differentiable tokamak transport simulator in JAX. Preprint at https://doi.org/10.48550/arXiv.2406.06718 (2024).
77. Klimesch, J., Drori, Y., Adhikari, R. X. & Krenn, M. Differometor: a differentiable interferometer simulator for the computational design of gravitational wave detectors. *GitHub* https://github.com/artificial-scientist-lab/Differometor (2025).
78. El-Kishky, A. et al. Competitive programming with large reasoning models. Preprint at https://doi.org/10.48550/arXiv.2502.06807 (2025).
79. Novikov, A. et al. AlphaEvolve: a coding agent for scientific and algorithmic discovery. Preprint at https://doi.org/10.48550/arXiv.2506.13131 (2025).
80. Moses, W. & Churavy, V. Instead of rewriting foreign code for machine learning, automatically synthesize fast gradients. *Adv. Neural Inf. Process. Syst.* **33**, 12472–12485 (2020).
81. Sagebaum, M., Albring, T. & Gauger, N. R. High-performance derivative computations using CoDiPack. *ACM Trans. Math. Softw.* **45**, 1–26 (2019).
82. Sagan, D., Hoffstaetter, G., Signorelli, M. & Coxe, A. Bmad-Julia: a Julia environment for accelerator simulations including machine learning. In *Proc. 15th International Particle Accelerator Conference (IPAC'24)* 2562–2565 (2024).
83. Wan, J., Alamprese, H., Ratcliff, C., Qiang, J. & Hao, Y. JuTrack: a Julia package for auto-differentiable accelerator modeling and particle tracking. *Comput. Phys. Commun.* **309**, 109497 (2025).
84. Aehle, M. et al. Optimization using pathwise algorithmic derivatives of electromagnetic shower simulations. *Comput. Phys. Commun.* **309**, 109491 (2025).
85. Aehle, M., Blühdorn, J., Sagebaum, M. & Gauger, N. R. Forward-mode automatic differentiation of compiled programs. *ACM Trans. Math. Softw.* **51**, 1–25 (2025).
86. Aehle, M. et al. Efficient forward-mode algorithmic derivatives of Geant4. Preprint at https://doi.org/10.48550/arXiv.2407.02966 (2024).
87. Christodoulou, S. & Naumann, U. Differentiable programming: efficient smoothing of control-flow-induced discontinuities. Preprint at https://doi.org/10.48550/arXiv.2305.06692 (2023).
88. Arya, G., Schauer, M., Schäfer, F. & Rackauckas, C. Automatic differentiation of programs with discrete randomness. *Adv. Neural Inf. Process. Syst.* **35**, 10435–10447 (2022).
89. Aehle, M. et al. Exploration of differentiability in a proton computed tomography simulation framework. *Phys. Med. Biol.* **68**, 244002 (2023).
90. Silvano, C. et al. A survey on deep learning hardware accelerators for heterogeneous HPC platforms. *ACM Comput. Surv.* **57**, 1–39 (2025).
91. Li, M. et al. The deep learning compiler: a comprehensive survey. *IEEE Trans. Parallel Distrib. Syst.* **32**, 708–727 (2020).
92. Aycock, J. A brief history of just-in-time. *ACM Comput. Surv. (CSUR)* **35**, 97–113 (2003).
93. Bociort, F. Why are there so many system shapes in lens design? *Proc. SPIE* **7849**, 114–126 (2010).
94. Li, H., Xu, Z., Taylor, G., Studer, C. & Goldstein, T. Visualizing the loss landscape of neural nets. *Adv. Neural Inf. Process. Syst.* **31**, 6391 (2018).
95. Richardson, J. W., Pandey, S., Bytyqi, E., Edo, T. & Adhikari, R. X. Optimizing gravitational-wave detector design for squeezed light. *Phys. Rev. D* **105**, 102002 (2022).
96. Meyer, J. J., Borregaard, J. & Eisert, J. A variational toolbox for quantum multi-parameter estimation. *npj Quantum Inf.* **7**, 89 (2021).
97. O'Driscoll, L., Nichols, R. & Knott, P. A. A hybrid machine learning algorithm for designing quantum experiments. *Quantum Mach. Intell.* **1**, 5–15 (2019).
98. Dorigo, T. et al. On the codesign of scientific experiments and industrial systems. Preprint at https://doi.org/10.48550/arXiv.2603.26613 (2026).
99. Nichols, R., Mineh, L., Rubio, J., Matthews, J. C. & Knott, P. A. Designing quantum experiments with a genetic algorithm. *Quantum Sci. Technol.* **4**, 045012 (2019).
100. Bindel, D., Landreman, M. & Padidar, M. Understanding trade-offs in stellarator design with multi-objective optimization. *J. Plasma Phys.* **89**, 905890503 (2023).
101. Ruiz, F. J. et al. Quantum circuit optimization with AlphaTensor. *Nat. Mach. Intell.* **7**, 374–385 (2025).
102. Zen, R., Nägele, M. & Marquardt, F. Reusability report: optimizing T count in general quantum circuits with AlphaTensor-Quantum. *Nat. Mach. Intell.* **8**, 113–117 (2025).
103. Venugopalan, G., Salces-Cárcoba, F., Arai, K. & Adhikari, R. X. Global optimization of multilayer dielectric coatings for precision measurements. *Opt. Express* **32**, 11751–11762 (2024).
104. McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **9**, 4812 (2018).
105. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *Proc. International Conference on Learning Representations (ICLR)* (2015).
106. Krenn, M., Kottmann, J. S., Tischler, N. & Aspuru-Guzik, A. Conceptual understanding through efficient automated design of quantum optical experiments. *Phys. Rev. X* **11**, 031044 (2021).
107. Landreman, M. et al. SIMSOPT: a flexible framework for stellarator optimization. *J. Open Source Softw.* **6**, 3525 (2021).
108. Strong, G. C. et al. TomOpt: differential optimisation for task- and constraint-aware design of particle detectors in the context of muon tomography. *Mach. Learn. Sci. Technol.* **5**, 035002 (2024).
109. Zaher, Z. et al. Optimization of a cosmic muon tomography scanner for cargo border control inspection. *J. Appl. Phys.* **138**, 194903 (2025).
110. Hvarfner, C., Hellsten, E. O. & Nardi, L. Vanilla Bayesian optimization performs great in high dimensions. In *Proc. 41st International Conference on Machine Learning (ICML'24)* 20793–20817 (PMLR, 2024).
111. Giuliani, A. Direct stellarator coil design using global optimization: application to a comprehensive exploration of quasi-axisymmetric devices. *J. Plasma Phys.* **90**, 905900303 (2024).
112. Roussel, R. et al. Bayesian optimization algorithms for accelerator physics. *Phys. Rev. Accel. Beams* **27**, 084801 (2024).
113. Flam-Shepherd, D. et al. Learning interpretable representations of entanglement in quantum optics experiments using deep generative models. *Nat. Mach. Intell.* **4**, 544–554 (2022).
114. Olle, J., Zen, R., Puviani, M. & Marquardt, F. Simultaneous discovery of quantum error correction codes and encoders with a noise-aware reinforcement learning agent. *npj Quantum Inf.* **10**, 126 (2024).
115. Trenkwalder, L. M., López-Incera, A., Nautrup, H. P., Flamini, F. & Briegel, H. J. Automated gadget discovery in the quantum domain. *Mach. Learn. Sci. Technol.* **4**, 035043 (2023).
116. Sarra, L., Ellis, K. & Marquardt, F. Discovering quantum circuit components with program synthesis. *Mach. Learn. Sci. Technol.* **5**, 025029 (2024).
117. Kundu, A. & Sarra, L. Reinforcement learning with learned gadgets to tackle hard quantum problems on real hardware. *Commun. Phys.* **9**, 44 (2026).
118. Melnikov, A. A., Sekatski, P. & Sangouard, N. Setting up experimental Bell tests with reinforcement learning. *Phys. Rev. Lett.* **125**, 160401 (2020).
119. Valcarce, X., Sekatski, P., Gouzien, E., Melnikov, A. & Sangouard, N. Automated design of quantum-optical experiments for device-independent quantum key distribution. *Phys. Rev. A* **107**, 062607 (2023).
120. Landgraf, J., Peano, V. & Marquardt, F. Automated discovery of coupled-mode setups. *Phys. Rev. X* **15**, 021038 (2025).
This paper introduces autoscatter, an AI-driven discovery framework for interpretable wave-scattering devices that can find counterintuitive and simpler architectures than human designs.
121. Côté, G., Lalonde, J.-F. & Thibault, S. Deep learning-enabled framework for automatic lens design starting point generation. *Opt. Express* **29**, 3841–3854 (2021).
122. Fürrutter, F., Chandani, Z., Hamamura, I., Briegel, H. J. & Muñoz-Gil, G. Synthesis of discrete–continuous quantum circuits with multimodal diffusion models. *Mach. Learn. Sci. Technol.* **7**, 025065 (2026).
123. Zoccheddu, S., Qasim, S. R., Owen, P. & Serra, N. Large language models for physics instrument design. *Mach. Learn. Sci. Technol.* **7**, 045005 (2026).
124. Wang, H. et al. Efficient evolutionary search over chemical space with large language models. In *Proc. Thirteenth International Conference on Learning Representations (ICLR 2025)* (2025).
125. Lu, D., Malof, J. M. & Padilla, W. J. An agentic framework for autonomous metamaterial modeling and inverse design. *ACS Photonics* **12**, 6071–6080 (2025).
126. Barman, K. G. et al. Large physics models: towards a collaborative approach with large language models and foundation models. *Eur. Phys. J. C* **85**, 1066 (2025).
127. Foreback, M. et al. ECLIPSE: an Evolutionary Computation Library for Instrumentation Prototyping in Scientific Engineering. Preprint at https://doi.org/10.48550/arXiv.2601.05098 (2026).
128. Fasoli, A. et al. Computational challenges in magnetic-confinement fusion physics. *Nat. Phys.* **12**, 411–423 (2016).
129. Lauro, L. D. et al. Optimization methods for integrated and programmable photonics in next-generation classical and quantum smart communication and signal processing systems. *Adv. Opt. Photonics* **17**, 526–622 (2025).
130. Kostelecký, V. A. & Russell, N. Data tables for Lorentz and CPT violation. *Rev. Mod. Phys.* **83**, 11–31 (2011).
This paper presents a systematic extension of the standard model with hundreds of parameters, each of them being a potential objective for AI-driven experimental design.
131. Russell, N. E. Mining the Data Tables for Lorentz and CPT Violation. In *Proc. Eighth Meeting on CPT and Lorentz Symmetry* 82–85 (World Scientific, 2020).
132. Will, C. M. in *General Relativity and John Archibald Wheeler* 73–93 (Springer, 2010).
133. DeMille, D., Doyle, J. M. & Sushkov, A. O. Probing the frontiers of particle physics with tabletop-scale experiments. *Science* **357**, 990–994 (2017).
134. Hamilton, P. et al. Atom-interferometry constraints on dark energy. *Science* **349**, 849–851 (2015).
135. Pikovski, I., Vanner, M. R., Aspelmeyer, M., Kim, M. & Brukner, Č. Probing Planck-scale physics with quantum optics. *Nat. Phys.* **8**, 393–397 (2012).
136. Lo, A. et al. Acoustic tests of Lorentz symmetry using quartz oscillators. *Phys. Rev. X* **6**, 011018 (2016).
137. Arndt, M. & Hornberger, K. Testing the limits of quantum mechanical superpositions. *Nat. Phys.* **10**, 271–277 (2014).
138. Safronova, M. et al. Search for new physics with atoms and molecules. *Rev. Mod. Phys.* **90**, 025008 (2018).
139. Arlt, S., Gu, X. & Krenn, M. Towards autonomous quantum physics research using LLM agents with access to intelligent tools. Preprint at https://doi.org/10.48550/arXiv.2511.11752 (2025).
140. Baek, J., Jauhar, S. K., Cucerzan, S. & Hwang, S. J. ResearchAgent: iterative research idea generation over scientific literature with large language models. In *Proc. 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)* 6709–6738 (Association for Computational Linguistics, 2025).

141. Gu, X. & Krenn, M. Forecasting high-impact research topics via machine learning on evolving knowledge graphs. *Mach. Learn. Sci. Technol.* **6**, 025041 (2025).
142. Sourati, J. & Evans, J. A. Accelerating science with human-aware artificial intelligence. *Nat. Hum. Behav.* **7**, 1682–1696 (2023).
143. Cadena, S. A. et al. ConStellaration: a dataset of QI-like stellarator plasma boundaries and optimization benchmarks. *Adv. Neural Inf. Process. Syst.* **38**, 17351–17377 (2025).
144. Elitez, D. et al. ColliderML: the first release of an OpenDataDetector high-luminosity physics benchmark dataset. In *Machine Learning and the Physical Sciences Workshop at NeurIPS 2025* (2025).
145. Carleo, G. & Troyer, M. Solving the quantum many-body problem with artificial neural networks. *Science* **355**, 602–606 (2017).
146. De Regt, H. W. *Understanding Scientific Understanding* (Oxford Univ. Press, 2017).
147. Krenn, M. et al. On scientific understanding with artificial intelligence. *Nat. Rev. Phys.* **4**, 761–769 (2022).
148. Brown, D. D. et al. FINESSE. *Zenodo* https://doi.org/10.5281/zenodo.12662017 (2025).
149. The LIGO Scientific Collaboration et al. Advanced LIGO. *Class. Quantum Gravity* **32**, 074001 (2015).

Acknowledgements We thank H. de Regt for comments.

Author contributions J.K. and M. Krenn led the conceptualization, overall drafting and revision of the manuscript. S.A., C.R.-G., C.R., X.G., P.H., P.V., C.H., Y.D., R.A., M.A., M. Kagan and L.H. participated in discussions, contributed domain-specific technical expertise and provided feedback on the manuscript.

Funding J.K., S.A. and M. Krenn acknowledge support by the European Research Council (ERC) under the European Union's Horizon Europe research and innovation programme (ERC-2024-STG, 101165179, ArtDisQ) and from the German Research Foundation (DFG; EXC 2064/1, project 390727645). X.G. acknowledges the support from the DFG through SFB 1375 (NOA) and the Alexander von Humboldt Foundation. P.H. acknowledges support from the Austrian Science Fund (FWF) Y1121, P36041, P35953 and the FFG-project AQUTEM. P.V. is supported by the 'Ramón y Cajal' programme under project no. RYC2021-033305-I funded by the MCIN MCIN/AEI/10.13039/501100011033, by the European Union NextGenerationEU/PRTR and by the European Innovation Council (EIC) Pathfinder project PHINDER, grant agreement no. 101258353, funded by the European Union. M.A. was supported in part by the Gordon and Betty Moore Foundation through grant no. 10771 (GBMF10771, https://doi.org/10.37807/GBMF10771). R.A. is supported by the WoodNext Foundation and grant no. 63405 from the John Templeton Foundation. M. Kagan is supported by the US Department of Energy (DOE) under grant no. DE-AC02-76SF00515. L.H. is supported by the Excellence Cluster ORIGINS, which is funded by the DFG under Germany's Excellence Strategy EXC-2094-390783311 and under the Collaborative Research Center SFB1258 grant - SFB-1258 - 283604770. The other authors received no specific funding for this work.

Competing interests J.K., S.A. and M. Krenn are in the process of founding Feyer GmbH, a company focused on AI for industrial and technological invention and have been selected for funding through the SPRIND Next Frontier AI Challenge. This activity began after the submission of this manuscript and no funding or company activity contributed to the work described in this Review. The other authors declare no competing interests.

Additional information
Correspondence and requests for materials should be addressed to Jonathan Klimesch or Mario Krenn.
Peer review information *Nature* thanks the anonymous reviewers for their contribution to the peer review of this work.
Reprints and permissions information is available at http://www.nature.com/reprints.
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

© Springer Nature Limited 2026